Universidad Nacional de La Plata

Facultad de Informática

Sistemas Recomendadores aplicados en Educación

Lic. María Emilia Charnelli

Director: Lic. Javier Díaz

Co-Directora: Dra. Laura Lanzarini

Trabajo presentado para obtener el grado de

Especialista en Tecnología Informática Aplicada en Educación

Septiembre de 2019

Índice general

Índice de figuras

Índice de tablas

Índice de acrónimos

AMT Aprendizaje Mejorado por Tecnología. 34

CB Recomendadores basados en el Contenido. 17, 19, 20, 23

CS similitud del Coseno. 27

DM Minería de Datos. 16

EVEA Entornos Virtuales de Enseñanza y Aprendizaje. 8, 35, 63

FC Filtrado Colaborativo. 17, 18, 19, 20, 21, 23, 25, 27, 30, 32, 38, 40, 41, 42, 70, 71

FCP Fracción de Pares Concordantes. 4, 49, 50, 52, 53, 54, 55, 65, 66, 67, 68

IR Recuperación de la Información. 16, 44

kNN k-Nearest Neighbors. 23, 24, 25, 26, 65, 66, 67, 68

LA Analítica del Aprendizaje. 36

MAE Error Absoluto Medio MAE. 4, 48, 49, 50, 51, 52, 55, 65, 66, 67, 68

MOOC Massive Open Online Courses. 36

MSE Error Cuadrático Medio. 47

OA Objetos de Aprendizaje. 7, 35, 39, 40, 41, 42

OAI Iniciativa de Archivos Abiertos. 39

REA Recursos Educativos Abiertos. 7, 62

RMSE Raíz del Error Cuadrático Medio. 4, 31, 48, 50, 51, 52, 55, 65, 66, 67, 68

SR Sistemas Recomendadores. 7, 8, 9, 12, 13, 14, 15, 16, 17, 19, 27, 33, 35, 36, 37, 42, 44, 45, 46, 47, 56, 57, 70, 72

SRE SR aplicados en Educación. 33, 34, 36, 37, 39, 42, 70

SVD Descomposición de Valores Singulares. 22, 30, 65, 67, 68

Capítulo 1

Introducción

1.1. Motivación

En la actualidad, tanto profesores como alumnos recurren a Internet para encontrar recursos que permitan complementar el proceso de enseñanza-aprendizaje. Sin embargo, ante la gran cantidad de oferta de material de estudio disponible, se dificulta la tarea de buscar y encontrar recursos que sean relevantes a sus necesidades. Los Sistemas Recomendadores (SR) son utilizados con el fin de facilitar esta tarea. Los contenidos personalizados que pueden proporcionar los SR incluyen: recursos educativos, como Recursos Educativos Abiertos (REA), Objetos de Aprendizaje (OA) y otros recursos web; posibles cursos a realizar; grupos de estudio; entre otros [1] [2] .

Generalmente, los SR aplicados en el ámbito educativo se centran en los estudiantes como el principal consumidor de recursos [3] y no consideran las preferencias de los docentes [4]. Es por esto que un algoritmo recomendador debe poseer la capacidad de adaptarse automáticamente a los objetivos educacionales de los alumnos haciéndolos coincidir con los objetivos educacionales propuestos por los docentes [5]. Para poder llevar adelante esta tarea, estos sistemas necesitan contar con la mayor cantidad de información posible de los usuarios para proveerles una recomendación razonable. El perfil de usuario es una

colección de información personal que en el ámbito educativo puede incluir desde intereses, preferencias e interacciones de los alumnos o docentes con el sistema, hasta estilos de aprendizaje y habilidades cognitivas de los alumnos.

Las Unidades Académicas cuentan con una gran cantidad de información valiosa de los alumnos y los docentes almacenada en los sistemas informáticos que utilizan. Estos sistemas incluyen tanto a los Entornos Virtuales de Enseñanza y Aprendizaje (EVEAs) como a los sistemas de gestión. Integrar la información provista por todos estos sistemas para construir un algoritmo recomendador preciso es una tarea compleja y generalmente toda la información no se encuentra disponible. En la actualidad existen otros ámbitos en los que los usuarios interactúan con asiduidad. La inclusión de información de otras fuentes como por ejemplo las redes sociales pueden otorgar una visión integral detectando similitudes y diferencias entre alumnos [6] [7].

Por otro lado, existen una gran variedad de repositorios de recursos educativos como Merlot y OER Commons que brindan sobre cada recurso descripciones, valoraciones, comentarios y metadatos que permiten conocer: título, objetivo didáctico, competencias, tipo de material (diapositivas, apunte, objeto de aprendizaje, etc.), para qué nivel educativo está dirigido, idioma, entre otras características. Esta información adicional puede ser utilizada por los SR para modelar y caracterizar a estos recursos, y garantizar la calidad de las sugerencias que se realicen.

La aplicación de SR en el ámbito educativo beneficia tanto a los educadores como a los alumnos. Por un lado, reducen la necesidad de los docentes de buscar y seleccionar materiales educativos; por otro lado, mejoran el proceso de toma de decisiones de los alumnos dado que pueden acceder a materiales de calidad e información precisa y personalizada.

1.2. Objetivos

Los SR analizan patrones de interés del usuario para proporcionar recomendaciones de artículos o productos que satisfagan sus preferencias y necesidades. Las sugerencias intervienen en varios procesos de toma de decisiones, tales como la compra de artículos o la elección de libros o películas [8]. La realización de recomendaciones personalizadas es un tema de investigación muy popular en diferentes dominios como el comercio electrónico, el entretenimiento y la educación.

Los SR aplicados en el ámbito educativo ofrecen una solución al problema de recuperar materiales ajustados al perfil de los alumnos y docentes. En este escenario, los SR permiten buscar recursos en uno o varios repositorios, y sugieren aquellos que mejor se adaptan no sólo a la búsqueda, sino también al perfil o necesidades educativas del individuo [3]. Para llevar adelante esta tarea pueden tener en cuenta: palabras claves, objetivos y/o estilos de enseñanza (docentes) y de aprendizaje (estudiantes), metadatos y descripciones de los recursos, entre otros.

El objetivo general de este trabajo es estudiar y analizar el estado del arte de los sistemas recomendadores y las modificaciones realizadas para el ámbito educativo. Esto comprende no solo las diferentes técnicas y algoritmos, sino también las métricas para evaluar la performance y la calidad de las recomendaciones.

Como objetivos específicos se trabajará en la recolección y el análisis de diferentes conjuntos de datos para recomendar recursos educativos, e implementar y comparar distintos algoritmos y métricas para evaluar las sugerencias obtenidas a partir de estos datos.

1.3. Publicaciones relacionadas a este trabajo

A continuación se detallan las publicaciones relacionadas a este trabajo:

- María Emilia Charnelli, Laura Lanzarini, Aurelio Fernández, Javier Díaz: New Item

Recommendation Method Based on Latent Topic Extraction. Proceedings in Business Analytics in Finance and Industry (BAFI). Universidad de Chile. 2018.

- María Emilia Charnelli, Laura Lanzarini, Javier Díaz: Recommender system based on latent topics. In: Argentine Congress of Computer Science, Communications in Computer and Information Science. Springer. 2017

- María Emilia Charnelli, Laura Lanzarini, Javier Díaz: Modeling students through analysis of social networks topics. In: XXII Argentine Congress of Computer Science (CACIC 2016). Selected Papers. 2016

- Laura Lanzarini, María Emilia Charnelli, Javier Díaz: Academic performance of university students and its relation with employment. Computing Latin American Conference CLEI. IEEE. 2015

- María Emilia Charnelli, Laura Lanzarini, Guillermo Baldino, Javier Díaz: Selección de atributos representativos del avance académico de los alumnos universitarios usando técnicas de visualización: Un caso de estudio. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología (TE & ET), vol 15, pp. 42-50. RedUNCI. 2015

- María Emilia Charnelli, Laura Lanzarini, Guillermo Baldino, Javier Díaz: Determining the profiles of young people from Buenos Aires with a tendency to pursue Computer Science studies. XX Argentine Congress of Computer Science. Selected papers. RedUNCI, pp. 66-74. 2015

1.4. Estructura del trabajo

A continuación se describe como se estructura este trabajo final integrador.

- **Sistemas Recomendadores**. En el capítulo 2 se presenta la definición y las técnicas de recomendación comúnmente utilizadas.

- **Sistemas Recomendadores aplicados en Educación**. En el capítulo 3 se desarrolla el estado del arte de los Sistemas Recomendadores aplicados en Educación y las técnicas de recomendación utilizadas.

- **Evaluación de los Sistemas Recomendadores**. En el capítulo 4 se explican las diferentes estrategias y métricas para evaluar a los Sistemas Recomendadores, en especial las métricas de evaluación comúnmente utilizadas y como se evalúan estos sistemas aplicados en el ámbito educativo.

- **Trabajo Experimental**. En el capítulo 5 se muestra el trabajo experimental realizado aplicando diferentes enfoques de recomendación en diferentes conjuntos de datos de dominio académico, donde se comparan los resultados utilizando métricas de validación.

- **Conclusiones y líneas de trabajo futuro**. Finalmente, en el capítulo 6 se presentan las conclusiones finales y las líneas de trabajo futuro.

Capítulo 2

Sistemas Recomendadores

2.1. Definición

Los SR se utilizan en una amplia variedad de aplicaciones, como la recomendación de libros, música, películas o noticias. La tarea de un SR es seleccionar automáticamente los artículos más apropiados para cada usuario de acuerdo a sus intereses y preferencias personales. Generalmente, un SR se centra en un tipo específico de elemento a recomendar denominado "ítem", como por ejemplo un e-commerce de viajes o un repositorio de materiales educativos, que genera recomendaciones personalizadas para proporcionar sugerencias útiles y efectivas para ese tipo específico de ítem [8].

El estudio de los SR es relativamente nuevo comparado con la investigación de otras herramientas y técnicas de los sistemas de información, como bases de datos o motores de búsqueda. Los SR surgieron como un área de investigación independiente a mediados de la década de 1990. Con el crecimiento y la variedad de información disponible en la Web y la rápida introducción de nuevos servicios de comercio electrónico como la compra de productos, la comparación de productos, entre otros, surgió la urgente necesidad de proporcionar recomendaciones. En los últimos años, ha aumentado el desarrollo de éstos sistemas en diferentes áreas como medicina, turismo, entretenimiento y educación.

Dependiendo del enfoque de recomendación, según el contexto y la necesidad del usuario, los SR generan sugerencias utilizando diversos tipos de conocimiento y datos sobre los usuarios, los ítems disponibles y transacciones anteriores almacenadas en bases de datos personalizadas. El usuario puede examinar las recomendaciones, puede aceptarlas o no y puede proveer, inmediatamente o en una próxima etapa, una retroalimentación implícita o explícita. Todas estas acciones del usuario y feedbacks se pueden almacenar en la base de datos del recomendador y se pueden utilizar para generar nuevas sugerencias en las próximas interacciones entre el usuario y el sistema [8].

Una transacción es una interacción registrada entre un usuario y el SR. La información generada a partir de las transacciones resulta de suma utilidad para el algoritmo recomendador. Por ejemplo, un registro de transacciones puede contener una referencia a un ítem seleccionado por el usuario y una descripción del contexto para esa recomendación particular. Si está disponible, esa transacción también puede incluir una retroalimentación explícita que el usuario haya proporcionado, como la calificación del elemento seleccionado. De hecho, las calificaciones son la forma más popular de datos de transacción que un SR recopila. Estas calificaciones se pueden recopilar explícita o implícitamente. En la recopilación explícita de calificaciones, se le pide al usuario que proporcione su opinión sobre un ítem en una escala de valores. Según [9], las calificaciones pueden adoptar una variedad de formas:

- Valoraciones numéricas del 1 al 5, que son una de las más utilizadas en aplicaciones.
- Valoraciones ordinales, como "totalmente de acuerdo, de acuerdo, neutral, en desacuerdo, totalmente en desacuerdo", donde el usuario selecciona el término que mejor indique su opinión con respecto a un ítem.
- Valoraciones binarias que modelan opciones para que el usuario decida si un determinado ítem le gustó o no.
- Valoraciones unarias que pueden indicar que un usuario ha observado o comprado

un ítem, o ha calificado el ítem positivamente. En tales casos, la ausencia de una calificación indica que no se tiene información que relacione al usuario con el ítem (tal vez compró el artículo en otro lugar).

Otra forma de evaluación consiste en que el usuario genere etiquetas para los elementos que presenta el sistema, por ejemplo poniendo alguna descripción de un artículo.

Para un SR, lo más conveniente es la retroalimentación explícita, donde los usuarios informan directamente sobre su interés en los productos a través de un puntaje o valoración. Debido a que la retroalimentación explícita no siempre está disponible, algunos recomendadores infieren las preferencias del usuario de la retroalimentación implícita, que indirectamente refleja la opinión a través de la observación del comportamiento del usuario. Los tipos de comentarios implícitos incluyen el historial de compras, el historial de navegación, los patrones de búsqueda o incluso los movimientos del mouse [10].

Herlocker et al. [11], define once tareas populares que un SR puede implementar.

- **Encontrar algunos ítems buenos**. El SR recomienda a un usuario una cantidad predefinida de elementos a través de una lista con predicciones de cuánto le gustaría al usuario (por ejemplo, en una escala de uno a cinco estrellas). Esta es la principal tarea de recomendación que abordan muchos sistemas comerciales y algunos sistemas no muestran la calificación pronosticada.

- **Encontrar todos los ítems buenos**. El SR sugiere todos los ítems que considere adecuados de acuerdo a las preferencias de los usuarios. En estos escenarios es insuficiente encontrar solo algunos artículos "buenos". Se utilizan en situaciones donde el número de ítems es relativamente pequeño o cuando el SR es de misión crítica, como en aplicaciones médicas o financieras. Es importante que el usuario pueda examinar cuidadosamente todas las posibilidades y analizar la clasificación que el SR genera para poder elegir la mejor opción de acuerdo a sus necesidades.

- **Anotación en el contexto**. Dado un contexto existente, el SR elige los ítems de acuerdo a las preferencias a largo plazo del usuario. Por ejemplo, un SR de TV podría anotar qué programas que se muestran en la guía de programación electrónica serán de interés del usuario en el futuro.

- **Recomendar secuencias**. El SR en lugar de centrarse en la generación de una sola recomendación, genera una secuencia de ítems que son de interés en su conjunto. Por ejemplo, un SR que recomiende una serie de televisión o libros relacionados.

- **Recomendar un paquete**. El SR sugiere un grupo de ítems que se relacionan entre ellos para formar una única recomendación. Por ejemplo, un plan de viaje puede estar compuesto por varias atracciones, destinos y servicios de alojamiento que se encuentran en un área delimitada. Desde el punto de vista del usuario, estas diversas alternativas se pueden considerar y seleccionar como un único destino de viaje.

- **Sólo búsqueda**. La tarea del recomendador es ayudar al usuario a explorar los elementos que con mayor probabilidad caen dentro del alcance de los intereses del usuario para esa sesión de navegación específica. Esta es una tarea que también ha sido respaldada por técnicas de hipermedia adaptativas [12].

- **Encontrar recomendaciones creíbles**. Algunos usuarios no confían en los SR, por lo que juegan con ellos para ver qué tan buenos son en la formulación de sugerencias. Por lo tanto, algunos SR también pueden ofrecer funciones específicas para permitir que los usuarios prueben su comportamiento además de los que se requieren para obtener recomendaciones.

- **Mejorar el perfil**. Esta tarea se relaciona con la capacidad del usuario de proveer información al SR sobre lo que le gusta y no le gusta. Esta es una tarea fundamental para poder proveer recomendaciones personalizadas. Si el sistema no tiene

conocimiento específico sobre el usuario activo, sólo podrá proveer a este usuario las mismas recomendaciones que le daría a un usuario "promedio".

- **Expresarse**. A algunos usuarios les resulta importante poder contribuir con sus calificaciones y expresar sus opiniones, y no tanto las recomendaciones que el SR les genera. La satisfacción del usuario en esta actividad permite mantener a este tipo de usuarios estrechamente vinculado con la aplicación, y darle más conocimiento al SR a través de sus opiniones que pueden ser utilizadas para las recomendaciones hacia otros usuarios.

- **Ayudar a otros**. Esta tarea está relacionada con la mencionada en el punto anterior. A algunos usuarios les gusta contribuir con sus puntajes porque creen que benefician a la comunidad con sus opiniones.

- **Influenciar en otros**. En relación a las dos tareas antes mencionadas, en los SR basados en la Web, hay usuarios cuyo objetivo principal es influir explícitamente en otros usuarios para que compren determinados productos. De hecho, también hay algunos usuarios malintencionados que pueden usar el sistema solo para promocionar o penalizar ciertos elementos.

Como indican estos diversos puntos, el rol de un SR dentro de un sistema de información puede ser bastante diverso. Esta diversidad requiere la explotación de diferentes fuentes de información y técnicas de conocimiento.

Los SR generalmente aplican técnicas y metodologías de otras áreas vecinas, como Interacción Persona-Computador (HCI, por sus siglas en inglés) y Recuperación de la Información (IR, por sus siglas en inglés). Sin embargo, en la mayoría de estos sistemas su algoritmo central se puede interpretar como una instancia de una técnica de Minería de Datos (DM, por sus siglas en inglés) [13]. Los procesos de Minería de Datos típicamente consisten en tres pasos, que se repiten sucesivamente: preparación de los datos, análisis de datos e interpretación de los resultados [14] [15].

2.2. Técnicas de recomendación

Los SR se pueden clasificar en cinco categorías diferentes dependiendo de la técnica de recomendación empleada para predecir los ítems de interés para un usuario:

- **Basados en el contenido (CB, por sus siglas en inglés)**: En ésta técnica el sistema aprende a recomendar ítems que son similares a otros que le gustaron en el pasado a un usuario en particular. La similitud entre ítems se calcula comparando características asociadas a los mismos. Por ejemplo, como se puede observar en la figura 2.1b, si un usuario calificó positivamente a un artículo que trata sobre la Historia del Arte, el sistema puede aprender a recomendar otros artículos que sean de la misma temática.

- **Filtrado Colaborativo (FC)**: El SR recomienda a un usuario activo los ítems que les gustaron a otros usuarios con similares preferencias. La similitud de gustos entre dos usuarios se calcula teniendo en cuenta la similitud entre la historia de calificaciones de los mismos. Dentro de los FC, los métodos basados en vecinos más próximos tienen gran popularidad por su simpleza, eficiencia, y su habilidad para producir recomendaciones precisas y personalizadas. Su forma original, esquematizada en la figura 2.1a, está basada en las similitudes entre usuarios. Dichos métodos de usuario-usuario estiman calificaciones desconocidas basadas en calificaciones registradas de usuarios con ideas afines. Posteriormente, se hizo popular el enfoque análogo pero ahora teniendo en cuenta las similitudes entre ítems [16]. Existen diferentes extensiones del FC, como las que utilizan modelos de factores latentes y métodos de factorización de matrices. Estos modelos intentan explicar las calificaciones representando tanto a los ítems como a los usuarios en un número de características numéricas, llamadas factores latentes, que son inferidas automáticamente a partir de la retroalimentación del usuario. En el contexto de recomendación de películas, por ejemplo, los factores descubiertos pueden medir dimensiones ob-

vias como que tan romántica es una película, la cantidad de acción que tiene o si es orientada o no a niños. Por lo general, también, se obtienen otros factores latentes menos interpretables. Los usuarios son caracterizados con otro conjunto de factores que miden el nivel de interés del usuario para cada factor latente de los ítems a recomendar.

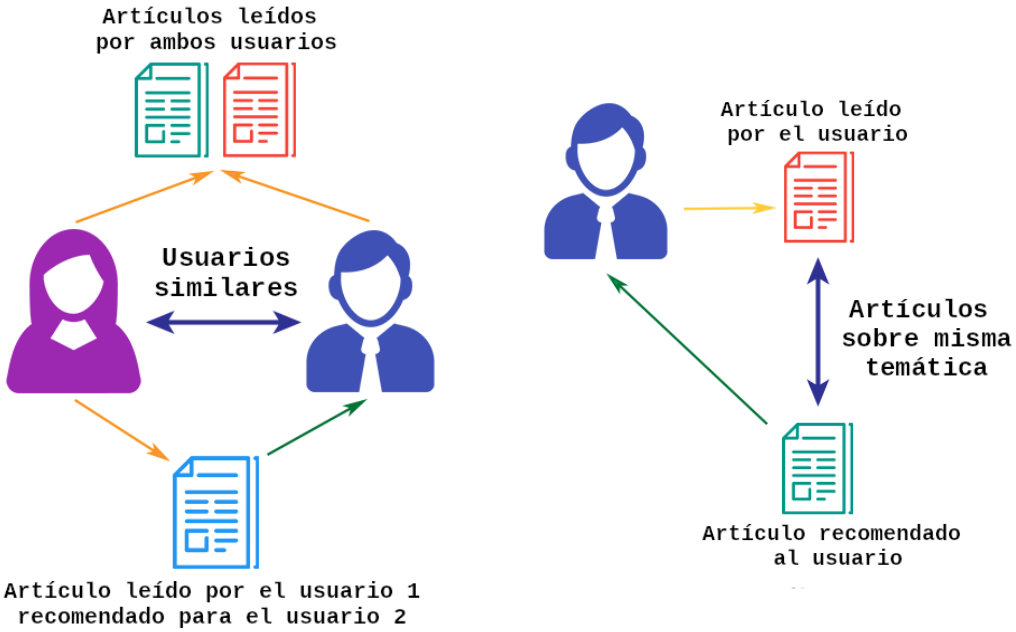
- **Demográfico**: Este tipo de sistemas recomiendan ítems teniendo en cuenta el perfil demográfico del usuario. La suposición es que diferentes usuarios deberían generar diferentes nichos demográficos. Muchos sitios web adoptan la personalización de soluciones efectivas y simples basándose en datos demográficos. Por ejemplo, usuarios que son redireccionados a sitios web particulares según su lenguaje o país. O sugerencias que pueden ser adaptadas de acuerdo a la edad de un usuario [10].

- **Basados en conocimiento**: Los sistemas basados en conocimiento recomiendan ítems teniendo en cuenta el dominio específico de conocimiento acerca de ciertas características de los ítems y las necesidades y preferencias de los usuarios [8]. La información de los usuarios como qué producto está buscando podría ser recolectada por ejemplo a través de un formulario de búsqueda, para luego, a través reglas de asociación definidas a priori, seleccionar los ítems que se ajusten al perfil de un usuario. Recolectar las reglas de asociación entre los ítems para calificarlos como similares o definir reglas entre los datos que se tengan de los usuarios y las características de los objetos a recomendar puede ser una tarea compleja a medida que crece la cantidad de ítems y usuarios en el sistema. Los sistemas basados en el conocimiento tienden a funcionar mejor que otros al comienzo de su implementación, pero si no están equipados con componentes de aprendizaje automático, pueden ser superados por las técnicas de FC.

- **Basados en comunidad**: Este tipo de sistemas recomienda ítems basándose en las preferencias de los amigos de los usuarios [17]. La evidencia sugiere que las personas

tienden a confiar más en las recomendaciones de sus amigos que en las recomendaciones de individuos similares pero anónimos. Esta observación, combinada con la creciente popularidad de las redes sociales abiertas, ha generado un creciente interés en los sistemas basados en la comunidad, también denominados SR sociales. Este tipo de SR modela y adquiere información sobre las relaciones sociales de los usuarios y las preferencias de los amigos del usuario. En general, las recomendaciones basadas en redes sociales no son más precisas que las derivadas de los enfoques tradicionales de FC, excepto en casos especiales, como cuando las calificaciones de los usuarios de un elemento específico son muy variadas o para situaciones de arranque en frío, es decir, donde los usuarios no proporcionaron calificaciones suficientes para calcular la similitud con otros usuarios [18].

- **Híbridos**: Estos SR se basan en la combinación de las técnicas mencionadas anteriormente. Un sistema híbrido que combina las técnicas A y B utiliza las ventajas de A para corregir las desventajas de B. Por ejemplo, los métodos FC sufren el problema de no poder recomendar ítems que no tienen calificaciones. Esto no limita a los enfoques CB, ya que la predicción de los nuevos elementos se basa en su descripción (características) que suelen ser fácilmente disponibles. Dadas dos o más técnicas de SR básicas, se han propuesto varias formas para combinar y crear un nuevo sistema híbrido.



(a) Filtrado Colaborativo (b) Basados en el contenido

Figura 2.1: Comparación entre técnicas de recomendación

El Filtrado Colaborativo es la técnica más popular y la más ampliamente utilizada por su efectividad y precisión [19] [20]. Por esta razón en este trabajo final integrador se analizarán y utilizarán sus diferentes enfoques.

2.3. Filtrado Colaborativo

Desde sus inicios, la técnica de Filtrado Colaborativo ha sido popular. El FC se basa únicamente en las valoraciones otorgadas por los usuarios y no depende de la disponibilidad de información adicional para la generación de recomendaciones, como es el caso de los recomendadores CB o los demográficos. Incluso con grandes volúmenes de datos, el FC puede generar recomendaciones eficaces y superiores que las obtenidas por un sistema CB [21].

Para poder generar recomendaciones, los algoritmos de FC necesitan relacionar dos entidades fundamentalmente diferentes: usuarios e ítems. Existen dos enfoques principales para facilitar dicha comparación, que constituyen las dos técnicas principales de FC: el enfoque de vecindad y los modelos de factores latentes.

El problema de recomendación puede definirse como la estimación de la respuesta de un usuario a nuevos elementos, en función de la información histórica almacenada en el sistema, y la sugerencia de elementos novedosos y originales para el usuario para los cuales la probabilidad de que la respuesta coincida con la respuesta previa es alta. Dado el conjunto de usuarios en el sistema U y el conjunto de elementos I, se define R como el conjunto de calificaciones registradas en el sistema, y S como el conjunto de valores posibles para una calificación, por ejemplo, $S = [1, 5]$ o $S = [\text{me gusta}, \text{no me gusta}]$. Se hace la suposición de que ningún usuario puede hacer más de una calificación en U para un artículo en particular $i \in I$ y la calificación de un usuario u para un ítem i se define como r_{ui}. Para identificar el subconjunto de usuarios que han calificado un ítem i se utiliza la notación U_i. Del mismo modo, I_u representa el subconjunto de elementos que han sido calificados por un usuario u. Los ítems que han sido calificados por dos usuarios u y v, es decir, $I_u \cap I_v$, se define como I_{uv}. De manera similar, U_{ij} se utiliza para definir el conjunto de usuarios que han calificado ambos elementos i y j. Para un determinado usuario u_a, denominado usuario activo, la tarea del algoritmo de FC de encontrar los ítems de más interés se puede calcular de dos maneras:

- **predicción**: Devuelve un valor numérico $\hat{r}_{u_a i}$ que expresa la predicción del puntaje obtenido para el elemento i dado el usuario u_a.
- **recomendaciones Top-N**: Devuelve una lista de N ítems, que serán los ítems que más le gustarán al usuario u_a.

Los métodos basados en vecindad son ampliamente utilizados debido a su simplicidad, su eficiencia y su capacidad para producir recomendaciones precisas y personalizadas.

Su forma original se basa en el enfoque usuario-usuario. En este enfoque se estiman calificaciones desconocidas basadas en calificaciones registradas de usuarios con ideas afines. Años más tarde se hizo popular el enfoque ítem-ítem. En estos métodos se calcula un calificación utilizando valoraciones realizadas por el mismo usuario en ítems similares. Una mejor escalabilidad y una precisión mejorada hacen que el enfoque por ítems sea más favorable en muchos casos [21]. Además, los métodos ítem-ítem son más susceptibles de explicar el razonamiento detrás de las predicciones. Esto se debe a que los usuarios están familiarizados con los elementos previamente preferidos por ellos, pero no conocen a los usuarios supuestamente parecidos.

A diferencia de los sistemas basados en vecindad, que usan las calificaciones almacenadas directamente en la predicción, los enfoques basados en modelos de factores latentes usan estas calificaciones para aprender un modelo predictivo. La idea general es modelar las interacciones entre usuarios e ítems con los factores que representan las características latentes de los usuarios y los ítems en el sistema. Este modelo luego se entrena usando los datos disponibles, y luego se usa para predecir las calificaciones de los usuarios para los nuevos artículos. Dentro de este enfoque, una de las técnicas más utilizadas es la Descomposición de Valores Singulares (SVD, por sus siglas en inglés) [22].

En general, los modelos de factores latentes ofrecen una alta capacidad expresiva para describir diversos aspectos de los datos. Por lo tanto, tienden a proporcionar resultados más precisos que los modelos de vecindad. Sin embargo, la mayoría de los sistemas comerciales se basan en los modelos de vecindad. La prevalencia de los modelos de vecindad se debe en parte a su relativa simplicidad. Sin embargo, hay razones más importantes para que los sistemas en producción se adhieran a esos modelos. Primero, naturalmente brindan explicaciones intuitivas del razonamiento detrás de las recomendaciones, que a menudo mejoran la experiencia del usuario más allá de lo que puede lograr una precisión mejorada [8].

2.3.1. Modelos basados en vecindad

Los algoritmos basados en vecindad utilizan toda la información de calificaciones de usuarios sobre los ítems para generar una predicción. Se utilizan técnicas estadísticas para encontrar el conjunto de usuarios o ítems, conocidos como vecinos, cuya vecindad es establecida por la similitud que hay entre los usuarios o los ítems, según sea el enfoque. Una vez obtenido el vecindario se produce una predicción o una recomendación Top-N para el usuario activo u_a.

A diferencia de los Recomendadores basados en el Contenido (CB), que utilizan el contenido de los elementos previamente calificados por un usuario u, los enfoques de filtrado colaborativo se basan en las calificaciones de u y también las de otros usuarios en el sistema. La idea clave es que la calificación de u para un nuevo ítem es probable que sea similar a la de otro usuario v, si u y v han calificado otros ítems de manera similar. Del mismo modo, es probable que califique dos ítems i y j de manera similar, si otros usuarios han otorgado valoraciones similares a estos dos ítems. Los enfoques colaborativos superan algunas de las limitaciones de los CB. Por ejemplo, los artículos para los cuales el contenido no está disponible o es difícil de obtener aún se pueden recomendar a los usuarios a través de los comentarios de otros usuarios. Además, las recomendaciones basadas en FC se basan en la calidad de los elementos evaluados por otros usuarios, en lugar de confiar en el contenido que puede ser un mal indicador de la calidad. A diferencia de los sistemas CB, los filtros colaborativos pueden recomendar elementos con contenido muy diferente, siempre y cuando otros usuarios ya hayan mostrado interés por estos diferentes elementos.

Los métodos de recomendación basados en vecindad son conocidos como k-Nearest Neighbors (kNN), porque predicen la calificación $\hat{r}_{ui}$ de un usuario u para un nuevo artículo i utilizando las calificaciones de usuarios o ítems más similares, es decir los vecinos más próximos del usuario u o los vecinos más próximos del artículo i, según sea el enfoque utilizado.

kNN Básico

En el enfoque kNN básico se predice la calificación $\hat{r}_{ui}$ de un usuario u para un nuevo artículo i teniendo en cuenta las calificaciones de usuarios similares a u en el enfoque usuario-usuario o los ítems que tuvieron calificaciones similares a i en el enfoque ítem-ítem. Para encontrar los vecinos más próximos, es decir los más similares, de un usuario u o ítem i se utiliza una función $sim(x, y)$ que dado dos elementos x e y devuelve un valor en $\mathbb{R}$ que indica el valor de semejanza entre x e y.

En el enfoque usuario-usuario, se supone que para cada usuario $v \neq u$ se dispone un valor dado por $sim(u, v)$ que representa la similitud de preferencia entre u y v. Los k-vecinos más cercanos (kNN) de u, definidos por $N(u)$, son los k usuarios v con la mayor similitud $sim(u, v)$ a u. Sin embargo, solo los usuarios que hayan calificado al ítem i pueden ser usados en la predicción de $\hat{r}_{ui}$. Se define este conjunto de vecinos como $N_i(u)$. La calificación $\hat{r}_{ui}$ se puede estimar como la calificación promedio otorgada a i por estos vecinos:

- kNN usuario-usuario:

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i(u)} sim(u, v) r_{vi}}{\sum_{v \in N_i(u)} |sim(u, v)|} \tag{2.1}$$

En 2.1 se pondera la contribución de cada vecino por su similitud con u. Se utiliza $|sim(u, v)|$ en lugar de $sim(u, v)$ porque los pesos negativos pueden generar calificaciones fuera del rango permitido.

En el enfoque ítem-ítem, se supone que para cada ítem $i \neq j$ se dispone un valor dado por $sim(i, j)$ que representa la similitud entre i y j. Los k-vecinos más cercanos (kNN) de i, definidos por $N_u(i)$, son los k ítems calificados por el usuario u que son más similares al ítem i. La calificación pronosticada de u para i se obtiene como un promedio ponderado de las calificaciones otorgadas por u a los elementos de $N_u(i)$:

- kNN ítem-ítem:

$$\hat{r}_{ui} = \frac{\sum_{j \in N_u(i)} sim(i,j) r_{uj}}{\sum_{j \in N_u(i)} |sim(i,j)|} \tag{2.2}$$

Una alternativa a esta suma ponderada, es tener en cuenta el puntaje medio de cada usuario μ_u ó cada ítem μ_i:

- kNN Means usuario-usuario:

$$\hat{r}_{ui} = \mu_u + \frac{\sum_{v \in N_i(u)} sim(u,v)(r_{vi} - \mu_v)}{\sum_{v \in N_i(u)} |sim(u,v)|} \tag{2.3}$$

- kNN Means ítem-ítem:

$$\hat{r}_{ui} = \mu_i + \frac{\sum_{j \in N_i(i)} sim(u,v)(r_{ju} - \mu_j)}{\sum_{j \in N_i(i)} |sim(u,v)|} \tag{2.4}$$

Esta variante conocida como kNN Means permite que, si no se tienen suficientes vecinos cercanos, el algoritmo pueda retornar cómo mínimo el puntaje promedio otorgado por un usuario, o el puntaje promedio para un ítem.

kNN con baseline

Usualmente, la mayoría de las calificaciones son desconocidas porque típicamente un usuario evalúa solo a una pequeña porción de los elementos. A su vez, los datos típicos de FC exhiben grandes sesgos en las calificaciones brindadas por los usuarios a los elementos, porque algunos usuarios otorgan calificaciones más altas que otros, y porque algunos ítems reciben calificaciones más altas que otros. Para combatir el sobreajuste de los escasos datos de calificación y las tendencias de las calificaciones, se pueden utilizar modelos que tengan en cuenta éstos sesgos utilizando puntajes de referencia [22].

Sea μ el puntaje promedio de todas las calificaciones y sea $K = (u, i)$ el conjunto de los pares (u, i) de los que se conoce a r_{ui}. Una predicción de referencia para un puntaje r_{ui} desconocido se define como b_{ui}:

$$b_{ui} = \mu + b_u + b_i$$

Los parámetros b_u y b_i indican las desviaciones observadas del usuario u y el ítem i, respectivamente, del promedio. Para estimar b_u y b_i se puede resolver el problema de mínimos cuadrados.

$$\min_{b_u, b_i} \sum_{(u,i) \in K} (r_{ui} - \mu - b_u - b_i)^2 + \lambda_1 (\sum_u b_u^2 + \sum_i b_i^2)$$

El primer término $\sum_{(u,i) \in K} (r_{ui} - \mu - b_u - b_i)^2$ se esfuerza por encontrar los b_u y los b_i que se ajusten a las calificaciones dadas. El término de regularización $\lambda_1 (\sum_u b_u^2 + \sum_i b_i^2)$ evita el sobreajuste penalizando las magnitudes de los parámetros. Este problema de mínimos cuadrados se puede resolver de manera eficiente mediante el método de descenso por gradiente estocástico.

Para calcular la predicción del puntaje $\hat{r}_{ui}$ para un usuario u y un ítem i ajustando los efectos de los usuarios y los ítems utilizando la predicción de referencia:

- kNN Baseline usuario-usuario:

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{v \in N_i(u)} sim(u, v)(r_{vi} - b_{vi})}{\sum_{v \in N_i(u)} |sim(u, v)|} \tag{2.5}$$

- kNN Baseline ítem-ítem:

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in N_i(i)} sim(i, j)(r_{uj} - b_{uj})}{\sum_{j \in N_i(i)} |sim(i, j)|} \tag{2.6}$$

Métricas de similitud

Para las técnicas de FC basadas en vecindad es necesario utilizar una medida de similitud entre los ítems o los usuarios, según el enfoque que se utilice. Los pesos de similitud juegan un doble papel porque en primer lugar, permiten la selección de vecinos de confianza cuyas calificaciones se utilizan en la predicción, y en segundo lugar, porque proporcionan los medios para dar más o menos importancia a estos vecinos en la predicción. La elección de una métrica de similitud es uno de los aspectos más críticos de la construcción de un SR de FC basado en vecindad, ya que puede tener un impacto significativo tanto en su precisión como en su rendimiento, y elegir la mejor métrica no es una tarea sencilla [23] [24].

La similitud $sim(x, y)$ refleja la distancia, la correlación o el peso entre dos pares usuarios: u y v, o dos ítems: i y j. El valor de la similitud se puede interpretar directamente como la utilidad de la recomendación para un usuario [8]. Por ejemplo, si a un usuario le interesó un elemento, también le gustará un elemento similar. Asimismo, si a un usuario le gustó un elemento, a un usuario similar también le interesará. Por lo tanto, la efectividad de una recomendación depende de la capacidad de identificar elementos similares, llamados vecinos. A su vez, la calidad de la recomendación dependerá del número y el valor de las similitudes de los vecinos.

Existen muchos métodos diferentes para calcular la similitud o el peso entre usuarios o ítems. La correlación de Pearson y la similitud del Coseno son las métricas comúnmente utilizadas [8].

- **Similitud del Coseno**:

  La similitud del Coseno (CS, por sus siglas en inglés) entre dos documentos se puede medir representando cada documento como un vector de frecuencias de palabras y calculando el coseno del ángulo formado por los vectores de frecuencia. Este concepto se puede adoptar en el FC, que utiliza usuarios o elementos en lugar

de documentos y calificaciones en lugar de frecuencias de palabras. Formalmente, sean n la cantidad de ítems y m la cantidad de usuarios, y sea R la matriz de puntajes $m \times n$. La similitud entre dos usuarios, u y v, se define como el coseno de los vectores m-dimensionales correspondientes a las filas u y v de la matriz R. Mientras que la similitud entre dos elementos, i y j, se define como el coseno de los vectores n-dimensionales correspondientes a las columnas i y j de la matriz R.

- Enfoque usuario-usuario:

  Se consideran los usuarios u y v como vectores x_u, x_v, donde $x_{ui} = r_{ui}$ si el usuario u ha calificado el artículo i, y 0 de lo contrario; de manera análoga si $x_{vi} = r_{vi}$. Sean $i \in I_{uv}$ todos los ítems que han sido evaluados tanto por el usuario u como por el usuario v. La similitud entre dos usuarios u y v se calcula de la siguiente manera:

$$CV(u,v) = \frac{\sum\limits_{i \in I_{uv}} r_{ui} r_{vi}}{\sqrt{\sum\limits_{i \in I_{uv}} r_{ui}^2 \sum\limits_{i \in I_{uv}} r_{vi}^2}}$$

- Enfoque ítem-ítem:

  De manera similar, se consideran los ítems i y j como vectores x_i, x_j, donde $x_{ui} = r_{ui}$ si el ítem i ha sido calificado por el usuario u, y 0 de lo contrario; de forma análoga si $x_{uj} = r_{uj}$. Sean $u \in U_{ij}$ todos los usuarios que han calificado tanto al ítem i como al ítem j. La similitud entre dos ítems i y j se calcula de la siguiente manera:

$$CV(i,j) = \frac{\sum\limits_{u \in U_{ij}} r_{ui} r_{uj}}{\sqrt{\sum\limits_{u \in U_{ij}} r_{ui}^2 \sum\limits_{u \in U_{ij}} r_{uj}^2}}$$

Si los vectores apuntan en la misma dirección, su similitud de coseno es 1.

Un problema con esta medida es que no considera las diferencias en la media y la varianza de las calificaciones. En una situación real, diferentes usuarios pueden

usar escalas de calificación diferentes, que la similitud del coseno no puede tener en cuenta.

- **Correlación de Pearson**:

  La correlación de Pearson mide el grado en que dos variables se relacionan linealmente entre sí. La correlación de Pearson cuantifica la fuerza y la dirección de la relación lineal entre dos variables aleatorias. Es similar al coseno, excepto que normaliza las calificaciones de ambos vectores en relación con su media. Debido a que esta similitud considera la diferencia en las escalas de calificación, por lo general es mejor que la similitud del coseno, que usa las calificaciones sin normalizar [25].

  - Enfoque usuario-usuario:

    $$PC(u,v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \mu_u)(r_{vi} - \mu_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \mu_u)^2 \sum_{i \in I_{uv}} (r_{vi} - \mu_v)^2}}$$

    donde μ_u es la calificación promedio que ha dado un usuario u a los ítems que evaluó.

  - Enfoque ítem-ítem:

    La misma idea se puede utilizar para obtener similitudes entre dos ítems i y j:

    $$PC(i,j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \mu_i)(r_{uj} - \mu_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \mu_i)^2 \sum_{u \in U_{ij}} (r_{uj} - \mu_j)^2}}$$

    donde μ_i es la calificación promedio que ha recibido un ítem i por todos los usuarios que lo evaluaron.

El signo de un peso de similitud indica si la correlación es directa o inversa, su magnitud va de 0 a 1 y representa la fuerza de la correlación.

En el estudio realizado por Gunawardana et al. [23] se obtuvieron mejores resultados utilizando la similitud del coseno para el tipo de tareas que se realizaron.

2.3.2. Modelos de factores latentes

Los modelos de factores latentes abordan el FC con el objetivo de descubrir características latentes que expliquen las calificaciones observadas. Uno de los modelos más populares son los que utilizan una factorización de la matriz de calificaciones R, también conocidos como modelos basados en la Descomposición de Valores Singulares (SVD, por sus siglas en inglés). En los últimos años, éstos modelos han ganado popularidad por su precisión y escalabilidad. La factorización SVD es utilizada para identificar factores semánticos latentes [26].

SVD

Los modelos de factorización de matrices mapean a los usuarios e ítems en un espacio de factores latentes de dimensionalidad f, de modo que las interacciones usuario-ítem son modeladas como productos internos en ese espacio.

Cada ítem i está asociado con un vector $q_i \in R^f$, y cada usuario u está asociado con un vector $p_u \in R^f$. El producto punto resultante $q_i^T \dot{p}_u$ captura la interacción entre el usuario u y el elemento i , que por ejemplo podría ser el interés general del usuario en las características de un ítem. La calificación final se crea al agregar la calificación promedio de todos los ítems μ y las calificaciones de referencia que dependen solo del usuario b_u o del ítem b_i. Por lo tanto, la predicción de una calificación se define:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u \tag{2.7}$$

Para aprender los parámetros del modelo (b_u, b_i, p_u, q_i), se minimiza la suma de los errores al cuadrado entre la calificación real y la predicha utilizando un algoritmo de descenso de gradiente estocástico:

$$\min_{b*,q*,p*} \sum_{(u,i)\in K} (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda(b_u^2 + b_i^2 + ||q_i||^2 + ||p_u||^2) \tag{2.8}$$

Minimizar esta función objetivo, equivale a minimizar la raíz cuadrada de la suma de los errores al cuadrado (RMSE).

Para minimizar la función objetivo, primero se establecen valores iniciales para los parámetros de la ecuación que se quieren minimizar y luego se itera para reducir el error entre el valor predicho y el valor real corrigiendo el valor anterior por un pequeño factor. Para que el modelo sea capaz de generalizar y no sobreajustar los datos, se introduce un término de penalización en la ecuación a minimizar. Esto se representa mediante un factor de regularización λ multiplicado por la suma cuadrada de las magnitudes de los vectores q_i y p_u. La constante λ generalmente se determina mediante validación cruzada.

La minimización se realiza típicamente por descenso por gradiente estocástico o mínimos cuadrados alternantes. Las técnicas de mínimos cuadrados alternantes primeron fijan los p_u para resolver los q_i y luego, fijan los q_i para resolver los p_u. Si uno de estos se toma como una constante, el problema de optimización es cuadrático y se puede resolver de manera óptima [27]. Una optimización fácil de la pendiente de gradiente estocástica fue popularizada por Funk [28]. El algoritmo recorre todas las calificaciones en los datos de entrenamiento. Para cada calificación dada r_{ui}, se realiza una predicción $\hat{r}_{ui}$ y se calcula el error de predicción asociado $e_{ui} = r_{ui} - \hat{r}_{ui}$. Dado un r_{ui} del conjunto de entrenamiento, se modifican los parámetros moviéndose en la dirección opuesta al gradiente, produciendo:

- $b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda \cdot b_u)$
- $b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda \cdot b_i)$
- $q_i \leftarrow q_i + \gamma \cdot (e_{ui} \cdot p_u - \lambda \cdot q_i)$
- $p_u \leftarrow p_u + \gamma \cdot (e_{ui} \cdot q_i - \lambda \cdot p_u)$

Este algoritmo utiliza un factor llamado tasa de aprendizaje γ que determina la relación entre el valor anterior y el nuevo valor calculado después de cada iteración. Se puede esperar una mayor precisión dedicando tasas de aprendizaje diferentes para los parámetros γ y λ. Por lo que se recomienda emplear distintas tasas de aprendizaje para los sesgos del usuario, los sesgos de los elementos y los factores en sí mismos [26].

En su forma básica, la factorización de la matriz caracteriza tanto a los ítems como a los usuarios mediante vectores de factores inferidos a partir de los patrones de calificación de los ítems. La alta correspondencia entre los factores de ítem y de usuario conduce a la recomendación de un artículo a un usuario. Estos métodos ofrecen una precisión de predicción superior a otras técnicas de FC. Al mismo tiempo, ofrecen un modelo compacto de memoria eficiente, que puede ser entrenado relativamente fácil.

Capítulo 3

Sistemas Recomendadores aplicados en Educación

3.1. Estado del Arte

Los Sistemas Recomendadores son muy populares como un área de investigación y aplicación en diferentes dominios como el comercio electrónico, el entretenimiento, entre otros. Sin embargo, fue sólo hacia principios del año 2000 cuando aparecieron las primeras aplicaciones notables en el campo de la educación, ya que se consideraba que el trabajo relevante estaba relacionado con el área de los sistemas educativos adaptativos [3].

Los SR aplicados en Educación (SRE) pueden ser muy diversos. La mayoría de éstos sistemas se utilizan para sugerir recursos de aprendizaje o para sugerir secuencias de recursos [29] [30]. Otro tipo de SRE se utilizan para proporcionar asesoramiento a los estudiantes sobre cursos para inscribirse, ó tutores con los que pueden conectarse. Los SRE también pueden ayudar a los profesores o tutores a mejorar sus cursos o controlar sus recursos de aprendizaje [31] [32] [4]. En el área de investigación, los SRE se han estudiado principalmente como un medio para asistir automáticamente a usuarios en la búsqueda de recursos adecuados. A diferencia de los SR para productos o servicios, el contexto

de los SRE debe tener en cuenta las preferencias de alumnos y/o profesores de ciertos materiales o tópicos, y cómo este material puede ayudarlos a lograr sus objetivos [33] [4].

Los SRE ofrecen una solución al problema de recuperar materiales educativos ajustados al perfil educativo del destinatario. En este escenario, los SRE permiten buscar recursos o materiales educativos en uno o varios repositorios digitales, y sugieren aquellos que mejor se adaptan no sólo a la búsqueda, sino también al perfil o a las necesidades educativas del individuo [3]. Para llevar adelante esta tarea estos sistemas pueden tener en cuenta: palabras claves, objetivos, estilos de enseñanza y/o de aprendizaje, entre otros. Estos sistemas pueden desempeñar un papel educativo importante, considerando la variedad de recursos de aprendizaje que se publican en línea y los beneficios de la colaboración entre docentes y alumnos.

Los repositorios digitales con recursos educativos dan soporte a la formación de comunidades de aprendizaje en línea al proporcionar una plataforma para la colaboración. Los profesores, tutores y alumnos acceden a repositorios y buscan recursos de interés; y en muchos casos, también intercambian experiencias y opiniones [3]. En los últimos años, se han creado numerosos repositorios con materiales educativos digitales y los usuarios se enfrentan a una gran cantidad de recursos disponibles en la web. Por lo tanto, estos usuarios probablemente se beneficiarían de la orientación y los servicios en línea que los ayudarán a identificar recursos de aprendizaje adecuados a partir de una amplia variedad de opciones.

Por otro lado, el Aprendizaje Mejorado por Tecnología (AMT, por sus siglas en inglés) tiene como objetivo diseñar, desarrollar y evaluar innovaciones socio-técnicas para diversos enfoques de enseñanza y aprendizaje. Esto involucra a los estudiantes individuales, pero también a los grupos y a los procesos de gestión del conocimiento [1]. Por lo tanto, es un dominio que generalmente abarca las tecnologías de apoyo a las actividades de enseñanza y aprendizaje, incluidas las tecnologías de recomendación que facilitan la recuperación de los recursos de aprendizaje pertinentes [34].

La personalización de la educación se vuelve aún más importante con el uso creciente de Entornos Virtuales de Enseñanza y Aprendizaje (EVEAs), repositorios digitales de Objetos de Aprendizaje (OAs) o recursos educativos, sistemas de gestión de alumnos, entornos de aprendizaje personal y dispositivos para escenarios de aprendizaje móvil [35]. La adopción de enfoques de aprendizaje personalizados y especialmente de los SR hoy en día es razonable debido a la gran demanda de interpretación de datos que se almacenan en las instituciones educativas y en diferentes entornos o servicios virtuales. Casi todo el comportamiento y acciones de los estudiantes se almacenan en servidores de instituciones educativas [1]. Además, las actividades de aprendizaje tienen lugar en entornos virtuales que están compuestos de numerosas herramientas y sistemas. Por ejemplo, los EVEAs proporcionan acceso a recursos de aprendizaje, pero no aseguran que los profesores o estudiantes de un curso los utilicen. Generalmente, los estudiantes usan herramientas adicionales para colaborar o encontrar recursos, por ejemplo, en el caso de que el material de aprendizaje ofrecido en el entorno virtual no sea suficiente. Las situaciones de aprendizaje también se vuelven cada vez más complejas debido al hecho de que los enfoques pedagógicos se diferencian entre los procesos de aprendizaje formal e informal, ambos tienen requisitos diferentes para el ambiente de aprendizaje y, como tal, para la recomendación dentro del ambiente [36]. Los SR deben poder establecer un acuerdo entre las recomendaciones que son de interés para los alumnos y las que requiere el profesor. En consecuencia, la necesidad de cantidades masivas de datos sobre el usuario y sus actividades en todos sus entornos de aprendizaje es necesario para facilitar recomendaciones precisas.

La digitalización del aprendizaje y el crecimiento de los datos educativos han fomentado la investigación y el desarrollo de Sistemas Recomendadores [3]. El uso de tecnología en educación ha permitido que la recopilación de datos sea un proceso inherente de entrega de contenido educativo a los estudiantes. Esto significa que el análisis de la conducta de aprendizaje ya no se relaciona solo con estudios piloto representativos, sino con el uso

de toda la población estudiantil. Esta tendencia se ha acelerado incluso con la aparición de los Massive Open Online Courses (MOOC) [37] y la aparición del campo de la Analítica del Aprendizaje (LA, por sus siglas en inglés) [38]. Los MOOC brindan cantidades masivas de datos de los estudiantes y, por lo tanto, brindan nuevas oportunidades para que los SR ofrezcan un apoyo de aprendizaje personalizado. La Analítica del Aprendizaje es actualmente un campo de investigación dentro de AMT que se enfoca en comprender y apoyar a los estudiantes en base a sus datos [1].

Existe una gran mayoría de SRE que tienen como objetivo encontrar contenido para apoyar las actividades de aprendizaje al proporcionarles nuevos materiales educativos. La segunda tarea de recomendación más utilizada es recomendar una secuencia de elementos a los alumnos. Recomendar una secuencia de elementos es una tarea muy importante dentro de un SRE porque es similar a los métodos de diseño instruccional. El objetivo de un diseño instruccional es guiar a un alumno a través de una serie de actividades de aprendizaje para lograr una cierta competencia. Este objetivo didáctico puede ser apoyado en el SR al sugerir los caminos más eficientes o eficaces a través de una lista de recursos de aprendizaje. Los SR con esta tarea a menudo consideran el conocimiento previo de un alumno para generar las sugerencias. La recomendación de alumnos pares también es una tarea de recomendación muy central para los entornos de educación a distancia y se aplica con relativa frecuencia en la investigación de SRE. Los estudiantes en línea generalmente se sienten aislados después de un período de tiempo si no tienen ninguna reunión presencial y por esta razón los cursos que se realizan enteramente a distancia tienden a tener mayores tasas de abandono en comparación con los cursos presenciales o semipresenciales [1]. Para superar esta situación, los SRE pueden ser de apoyo al recomendar a los alumnos formar parte de un equipo dentro de un curso en línea. En los últimos años, han aparecido nuevas tareas de recomendación como predecir el rendimiento del aprendizaje o sugerir una actividad de aprendizaje en contraste con un contenido de aprendizaje. Estos desarrollos muestran que los SR se aplican cada vez

más para filtrar y personalizar información en entornos de aprendizaje digital y también se aplican para nuevos objetivos educativos. [1]

En cuanto a las técnicas de recomendación utilizadas, los SRE suelen basarse en técnicas de filtrado colaborativo, filtrado basado en contenido, filtrado basado en conocimiento o algoritmos de recomendación híbridos. Estos algoritmos utilizan información sobre usuarios y recursos para generar las recomendaciones. La mayoría de los SRE se basan en perfiles de estudiantes o profesores que describen información adicional además de sus intereses y/o preferencias. El nivel de conocimiento del alumno en algunos casos es utilizado para personalizar las recomendaciones, tales como su conocimiento de los conceptos del curso o las calificaciones académicas pasadas. Los estilos de aprendizaje también son considerados por algunos recomendadores [39], a menudo basados en el inventario de Felder-Silverman [40].

Uno de los temas de investigación actual es la estandarización de los criterios de evaluación de los SRE. Además de los criterios de los SR tradicionales cómo la eficiencia y la precisión de las recomendaciones, éstos sistemas deberían considerar criterios específicos como la eficacia y la eficiencia del proceso de aprendizaje; la calidad de los materiales, el estilo de enseñanza y las necesidades de los docentes [3] [4].

Otra de las problemáticas dentro del campo de la investigación y a diferencia de otros dominios más populares como los de e-commerce, es que son muy pocos los conjuntos de datos de prueba disponibles para probar nuevas técnicas de recomendación. Además cada uno de éstos datos fueron utilizados para resolver problemáticas específicas por lo que disponen de diferente información acerca de los ítems, los usuarios y las valoraciones, lo que dificulta la comparación de diferentes métodos sobre diferentes conjuntos de datos que evalúan o disponen de otras fuentes de información para crear las recomendaciones [8] [36].

3.2. Filtrado Colaborativo en Educación

En términos de los métodos utilizados para la personalización de las recomendaciones, las técnicas más aplicadas son las basadas en filtrado colaborativo FC y las técnicas híbridas. Los primeros indicios de relacionar las técnicas de FC con la educación aparecieron en los trabajos de investigación de Terveen et al. [41], y Chislenko y Alexander [42].

Uno de los primeros intentos de desarrollar un recomendador de FC para recursos de aprendizaje fue el sistema Altered Vista [43]. El objetivo de este sistema era recopilar evaluaciones de recursos de aprendizaje proporcionados por los usuarios y recomendar estos recursos a usuarios similares. El equipo de Altered Vista exploró varios temas relevantes, como el diseño de su interfaz, el desarrollo de metadatos para almacenar evaluaciones proporcionadas por el usuario, el diseño y la arquitectura del sistema, y realizaron estudios pilotos y empíricos del uso del sistema para recomendar a los miembros de una comunidad recursos de su interés y sugerir conexiones con personas con preferencias similares.

Otro de los primeros sistemas fue CoFIND propuesto por Dron et al. [44]. El sistema utilizaba FC en combinación con datos de folksonomía para sugerir a estudiantes recursos relevantes que otros estudiantes habían encontrado previamente como valiosos. Los usuarios podían clasificar los recursos en uno o más temas y calificar a los mismos a través de metadatos pedagógicos establecidos por CoFIND. Este sistema sólo fue evaluado de forma teórica.

Otro sistema que se propuso para la recomendación de recursos de aprendizaje fue el Sistema de FC basado en Reglas (RACOFI) [45]. RACOFI combinaba dos enfoques de recomendación integrando un motor de FC que utilizaba las calificaciones que los usuarios proporcionaban para los recursos de aprendizaje complementados con un motor de reglas de inferencia que extraía las reglas de asociación entre los recursos de aprendizaje y los utilizaba para generar la recomendación. Pero los estudios de RACOFI no se implementaron ni se evaluaron en el ámbito académico. La tecnología RACOFI actualmente se utiliza en el sitio comercial inDiscovery para la recomendación de pistas de música.

En en el sistema QSIA desarrollado por Rafaeli et al. [46], el filtrado colaborativo tradicional se ampliaba con un mecanismo de control para marcar a los usuarios que deberían ser considerados para recomendaciones. Este sistema promovía la colaboración, la recomendación en línea y la formación posterior de las comunidades de aprendizaje. En lugar de desarrollar un sistema típico de recomendación automática, QSIA se basaba en un proceso de recomendación controlado principalmente por el usuario. Es decir, el usuario podía decidir si asumía el control sobre qué amigos aconsejar o si utilizaría un servicio de filtrado colaborativo. El sistema fue implementado y utilizado en el ámbito educativo pero no se reportaron resultados de evaluación.

Avancini y Straccia propusieron el sistema CYCLADES [47] que era un entorno donde los usuarios buscaban, accedían y evaluaban los recursos de aprendizaje disponibles en repositorios encontrados a través de la Iniciativa de Archivos Abiertos (OAI) [48]. Informalmente, OAI es un acuerdo entre varios proveedores de archivos digitales para ofrecer un nivel mínimo de interoperabilidad entre ellos. Por lo tanto, dicho sistema podía ofrecer recomendaciones sobre los recursos que se almacenaban en diferentes repositorios y se accedía a ellos a través de un esquema abierto. Las recomendaciones ofrecidas por CYCLADES fueron evaluadas a través de un estudio piloto con aproximadamente 60 usuarios que se centró en probar el rendimiento de varios algoritmos de filtrado colaborativo.

Shen y Shen [49] desarrollaron un SRE para OAs que se basaba en reglas de secuenciamiento que ayudaban a los usuarios a guiarse a través de los conceptos de una ontología de temas. Las reglas se activaban cuando se identificaban las deficiencias en las competencias de los alumnos, y luego se proponían los recursos apropiados para los mismos. Se llevó a cabo un estudio piloto con los estudiantes de una escuela de educación en línea donde los usuarios proporcionaron comentarios sobre el sistema.

Tang et al. [50] propusieron un sistema de e-learning que incluía un servicio de recomendación híbrido de recursos de aprendizaje que se podían encontrar en la web. Este sistema se encargaba de almacenar y compartir documentos de investigación y términos

de un glosario entre estudiantes universitarios y profesionales de la industria. Los recursos se describían a través de etiquetas de acuerdo con su contenido y aspectos técnicos, y los alumnos proporcionaban comentarios sobre éstos recursos en forma de calificaciones. La recomendación tenía lugar al involucrar un módulo de agrupación que utilizaba técnicas de agrupación de datos para establecer relaciones entre estudiantes con intereses similares y un módulo de FC para identificar a los alumnos con intereses similares en cada grupo. Los autores estudiaron varias técnicas para mejorar el rendimiento de su sistema como el uso de estudiantes simulados. También realizaron un estudio de evaluación del sistema con estudiantes reales.

Otro sistema que adopta un enfoque híbrido para recomendar recursos de aprendizaje fue el propuesto por Drachsler et al. [1]. Los autores se basaron en un estudio de simulación previo de Koper [51] para proponer un sistema que combinaba técnicas de recomendación basadas en la información social utilizando datos de otros alumnos y utilizando metadatos de perfiles y actividades de aprendizaje.

Una cantidad considerable de investigadores se han centrado en los criterios de múltiples atributos de recursos para cubrir la complejidad del aprendizaje como: conocimiento previo, experiencia, tiempo de estudio disponible, etc. Manouselis et al. [3] propusieron un algoritmo de FC para recomendar OAs considerando evaluaciones multidimensionales sobre los recursos proporcionadas por los profesores a través de una evaluación de pares. Sicilia et al. [52] investigaron las calificaciones multicriterio con OAs obtenidos del repositorio digital de materiales educativos MERLOT [53]. Salehi et al. [54] utilizaron un árbol de aprendizaje para tener en cuenta los múltiples atributos explícitos de los recursos, la variabilidad temporal del alumno y la matriz de calificaciones de los alumnos para un FC basado en atributos implícitos y explícitos. Tang et al. [50] consideraron las calificaciones multidimensionales de los OAs para relacionar usuarios.

También se han propuesto otros enfoques híbridos sobre los algoritmos de FC. En [55] se muestra que un algoritmo de FC basado en grafos puede mejorar la precisión de las

recomendaciones generadas incluso cuando los datos de las acciones del usuario son escasos. En [56] las técnicas de análisis de sentimientos sobre los comentarios generados por los usuarios de un repositorio de recursos educativos se utilizan para obtener información cualitativa valiosa para ajustar la calificación percibida de un recurso determinado por un usuario específico. En el sistema propuesto en [57] se utiliza un FC basado en usuarios que utiliza un algoritmo genético para optimizar la función de grado de interés de los estudiantes sobre los recursos. Este sistema además tiene en cuenta registros de actividades de aprendizaje realizados por los alumnos, como así también sus preferencias obtenidas por medio de un cuestionario, y tiene etiquetas que describen a los recursos educativos a sugerir. En este sistema, el profesor puede enviar los recursos al alumno por adelantado según el plan de enseñanza.

El FC a menudo comparte un terreno común con la navegación social, lo que también podría verse como un enfoque de recomendación social [58]. Fazeli et al. [55] mostró que la integración de la interacción social puede mejorar los enfoques de filtrado colaborativo en entornos educativos.

Investigadores como Manouselis, Verbet, Vuorikari y Kopeinik han realizado evaluaciones de técnicas de FC utilizando conjuntos de datos educativos para comparar y evaluar el resultado de las recomendaciones aplicando diferentes enfoques y métricas de similutud [3] [33]. Los datos de prueba que generalmente se utilizan son los que participaron del primer dataTEL challenge realizado en 2010 [59], donde estos datos educativos capturaban interacciones de estudiantes con herramientas y recursos educativos, algunos de ellos mencionados previamente como MACE [60], Travel Well [61] y Mendeley [62]. Otros autores han propuesto diferentes métricas de similitud para las técnicas de FC basadas en vecindad. Cómo se mencionó en el Capítulo 2, las métricas de similitud son utilizadas para comparar a los usuarios o a los ítems según el enfoque a utilizar, y la elección de esta métrica influirá en el resultado y la calidad de las sugerencias del recomendador. Niemann y Wolpers [63] investigaron el contexto de uso de OAs como una medida de

similitud para predecir y completar calificaciones de usuarios ausentes. El enfoque sugerido no requiere ninguna información de contenido de los OAs y, por lo tanto, también se puede aplicar para el arranque en frío de un enfoque basado en usuarios. Rojas et al. [24] presentan un análisis comparativo de las métricas de similitud basadas en dos criterios diferentes: la calidad y la cantidad de las recomendaciones proporcionadas. Las métricas se evaluaron en un SRE de FC que sugiere recursos a usuarios registrados en la Federación de Repositorios de Objetos de Aprendizaje de Colombia (FROAC).

Es importante destacar que la mayoría de los SRE mencionados son prototipos o ya no están en funcionamiento, mientras que los SRE actuales no comparten de forma abierta a la comunidad implementaciones ni conjuntos de datos parciales para probar nuevas técnicas en el área educativa [1]. Por otro lado, la disponibilidad de conjuntos de datos educativos de portales de recursos educativos como los dataTEL challenge han permitido evaluar y comparar distintas técnicas de SR en el ámbito educativo [36]. Sin embargo, actualmente la mayoría de los proyectos impulsores de este tema han finalizado y como consecuencia los datos de prueba no se encuentran disponibles o son díficiles de conseguir.

Capítulo 4

Evaluación de los Sistemas Recomendadores

4.1. Tipos de Experimentos

Cuando se decide utilizar un Sistema Recomendador en una aplicación se debe elegir entre una gran variedad de enfoques disponibles y se debe tomar una decisión sobre qué algoritmo será más apropiado según el contexto y los objetivos previamente establecidos. Comúnmente, estas decisiones se basan en experimentos que comparan la performance sobre un número de recomendadores candidatos. Generalmente se selecciona el algoritmo que tenga mejor rendimiento de acuerdo a la memoria disponible y a la capacidad de la CPU ó bien teniendo en cuenta las restricciones estructurales como el tipo, la confiabilidad y la disponibilidad de los datos.

En el campo de la investigación, cuando se proponen nuevos algoritmos de recomendación también se compara el rendimiento del nuevo algoritmo con un conjunto de enfoques existentes. Dichas evaluaciones se realizan generalmente aplicando una métrica de evaluación que proporciona una clasificación de los algoritmos candidatos, generalmente utilizando métricas de precisión.

Actualmente, si bien la precisión de las predicciones de los SR son importantes pueden no ser suficientes para implementar un buen motor de recomendación. Los usuarios también pueden estar interesados en descubrir o explorar rápidamente diversos artículos, en obtener respuestas rápidas del sistema y en muchas otras cuestiones que tienen que ver con la interacción con el motor de recomendación. Por lo tanto, se deben identificar el conjunto de condiciones o propiedades que el SR debe cumplir en el contexto de una aplicación específica. Y a partir de este punto, evaluar el sistema teniendo en cuenta estas propiedades relevantes para poder garantizar el éxito de un SR.

Generalmente, la evaluación de los Sistemas Recomendadores se realiza a través de tres tipos de experimentos que son motivados por la evaluación de protocolos en áreas como la Recuperación de la Información y el aprendizaje automático [64]:

- **Experimentos offline**, que utilizan un conjunto de datos previamente recolectados o simulados para testear la performance de los algoritmos candidatos;
- **Estudios de usuarios**, donde un pequeño grupo de usuarios realiza una serie de tareas utilizando el sistema en un ambiente controlado e informa sobre su experiencia;
- **Pruebas online**, donde un sistema es probado bajo diferentes condiciones durante su operación normal con sus usuarios reales. Tales experimentos evalúan el rendimiento de los recomendadores en usuarios que no están conscientes del experimento realizado.

En todos los escenarios de experimentación es importante considerar una serie de pasos:

1. *Hipótesis*: Antes de arrancar con el experimento se debe formular una hipótesis. Es importante definir una hipótesis que sea precisa y concisa y diseñar un experimento que pruebe la misma. Por ejemplo, una hipótesis puede ser que un algoritmo A

tenga una mejor predicción de puntajes de los usuarios que un algoritmo B. En este caso, el experimento debería evaluar la precisión de la predicción y no otros factores.

2. *Control de las variables*: Cuando se comparan un par de algoritmos candidatos sobre una hipótesis definida, es importante que todas las variables que no se prueben se mantengan fijas. Por ejemplo, si se quiere comparar la precisión de predicción entre dos algoritmos A y B, se debe entrenar a los algoritmos con los mismos conjuntos de datos, para poder entender si uno de los dos algoritmos tiene mejor performance que el otro.

3. *Poder de generalización*: Para aumentar la probabilidad de generalización de los resultados se debe experimentar con varios conjuntos de datos o aplicaciones. Es importante comprender las propiedades de los diversos conjuntos de datos que se utilizan y en términos generales, cuanto más diversa sea la información utilizada, mejor se podrán generalizar los resultados.

Este trabajo final se enfocará en los experimentos offline que son los menos costosos de llevar adelante y porque además los datos de prueba que se utilizarán no se probarán sobre un sistema recomendador que esté en línea.

4.2. Experimentos offline

Un experimento offline es evaluado utilizando un conjunto de datos de usuarios que eligen o evalúan a través de un puntaje a los ítems. Utilizando estos datos se intenta simular el comportamiento de los usuarios que interactúan con el SR. Este tipo de experimentos son atractivos porque no requieren de interacción real con los usuarios, y esto permite poder comparar una amplia variedad de algoritmos candidatos a muy bajo costo. La desventaja de los experimentos offline es que pueden responder a un conjunto muy

limitado de preguntas, generalmente preguntas sobre el poder de predicción de un algoritmo. Por lo que el objetivo de los experimentos offline es filtrar los enfoques inapropiados, dejando un conjunto relativamente pequeño de algoritmos candidatos para ser evaluados por los estudios de usuarios más costosos o los experimentos online. Un ejemplo típico de este proceso es cuando los parámetros de los algoritmos se ajustan en un experimento offline, y luego el algoritmo con los mejores parámetros continúa a la siguiente fase.

Como el objetivo de la evaluación offline es filtrar algoritmos, es importante asegurar que no haya ningún sesgo en las distribuciones de usuarios, ítems y evaluaciones seleccionadas. Por ejemplo, si se excluyeran los ítems o usuarios con pocas evaluaciones para reducir los costos de experimentación esto introducirá un sesgo sistemático en los datos. Si es necesario, el muestreo aleatorio de usuarios y artículos puede ser un método preferible para reducir los datos, aunque esto también puede introducir otros sesgos en el experimento, por ejemplo, esto podría favorecer a los algoritmos que funcionan mejor con datos dispersos. En ocasiones, los sesgos conocidos en los datos pueden corregirse mediante técnicas tales como volver a ponderar los datos, pero corregir los sesgos en los datos suele ser difícil. Otra fuente de sesgo puede ser la recolección de datos en sí misma. Por ejemplo, es más probable que los usuarios califiquen los ítems sobre los que tienen opiniones fuertes, y algunos usuarios pueden proporcionar muchas más calificaciones que otros. Por lo tanto, el conjunto de ítems puede estar sesgado por las propias calificaciones. Una vez más, las técnicas como el remuestreo o la reponderación de los datos de prueba pueden usarse para intentar corregir dichos sesgos.

4.3. Métricas de evaluación

Una vez definidas las propiedades relevantes para el SR de acuerdo a su contexto de aplicación, se pueden diseñar o modificar algoritmos que mejoren estas propiedades. Y al mejorar una propiedad se puede llegar a reducir la calidad de otra propiedad, e incluso

se puede afectar el rendimiento general del sistema. Por lo que se tienen que ejecutar experimentos adicionales o consultar las opiniones de los expertos de dominio.

Las métricas de precisión son las más utilizadas y pueden ser evaluadas con un análisis offline de los datos porque medirlas es típicamente independiente a la interfaz del usuario. En la base de la gran mayoría de éstos sistemas se encuentra un motor de predicción, y este motor puede predecir las opiniones de los usuarios sobre los ítems, por ejemplo, calificaciones de películas o la probabilidad de uso. Una suposición básica en un SR es que el usuario preferirá un sistema que proporcione predicciones más precisas. Y por ésta razón, se necesitan aplicar diferentes métricas en los algoritmos recomendadores para obtener los que brinden mejores predicciones.

A continuación se describirán métricas de precisión predictiva que evalúan la precisión del sistema comparando las calificaciones numéricas de las recomendaciones obtenidas. El SR genera predicciones de calificaciones $\hat{r}_{u,i}$ para un conjunto de datos de prueba τ que contiene $N = |\tau|$ pares de usuario-ítem (u, i) donde las verdaderas calificaciones $r_{u,i}$ se conocen. Generalmente, los $r_{u,i}$ se conocen porque se obtuvieron mediante un estudio de usuario, un experimento online u offline. Éstas métricas, también conocidas como métricas de error, permiten la evaluación de la calidad de la predicción numérica y sólo se puede realizar con ítems que hayan sido puntuados. Además, son utilizadas sólo para conjuntos de datos que no sean binarios, ya que miden que tan cercana es la predicción $\hat{r}_{u,i}$ al puntaje numérico real expresado por el usuario $r_{u,i}$.

4.3.1. MSE

El Error Cuadrático Medio (MSE, por sus siglas en inglés) adoptado en [65], es una métrica de error definida como:

$$MSE = \frac{1}{N} \sum_{(u,i)\in\tau} (\hat{r}_{u,i} - r_{u,i})^2 \tag{4.1}$$

MSE es muy fácil de computar, pero tiende a exagerar los efectos de posibles valores anómalos en las calificaciones.

4.3.2. RMSE

La Raíz del Error Cuadrático Medio (RMSE, por sus siglas en inglés) usada en [66], es una variación de MSE, que está dada por:

$$RMSE = \sqrt{\frac{1}{N} \sum_{(u,i)\in\tau} (\hat{r}_{u,i} - r_{u,i})^2} \tag{4.2}$$

donde la raíz cuadrada dada por MSE es la misma dimensión que para el valor predicho. Como MSE, RMSE eleva al cuadrado el error antes de sumarlo y sufre del mismo problema de outliers [9]. RMSE es la métrica más popular para evaluar la precisión de las calificaciones predichas.

4.3.3. MAE

El Error Absoluto Medio MAE (MAE, por sus siglas en inglés) es una métrica comúnmente utilizada entre calificaciones y predicciones. MAE es una medida de desviación de las recomendaciones a partir de valores verdaderos especificados por el usuario. Para obtener MAE, primero se suman los errores absolutos de los N pares de calificaciones reales y sus predicciones, y luego, se calcula el promedio de esa suma. A valor más pequeño de MAE, más preciso es el algoritmo recomendador para predecir las calificaciones de los usuarios.

$$MAE = \frac{1}{N} \sum_{(u,i)\in\tau} |\hat{r}_{u,i} - r_{u,i}| \tag{4.3}$$

MAE es la métrica más utilizada porque es de fácil implementación y directa interpretación [16].

4.3.4. MAE Normalizado

El Error Medio Absoluto Normalizado [67]. Está dado por:

$$NMAE = \frac{MAE}{\bar{r}_{max} - \bar{r}_{min}} \tag{4.4}$$

El valor de la métrica MAE da valores en un rango de 0 a ∞, y éste valor superior dependerá del rango de valores de las calificaciones, por ejemplo de 1 a 5 estrellas o una valoración del 1 al 10, tendrán diferentes rangos de valor. Para que este valor esté en un rango entre 0 y 1 se puede calcular el MAE Normalizado, donde r_{max} es el máximo puntaje posible y r_{min} es el mínimo puntaje posible.

Esta métrica, a diferencia de MSE, es menos sensible a outliers. Sin embargo, MAE no es siempre la mejor opción. Tanto MAE como métricas relacionadas pueden ser menos significativas para tareas como Finding Good Items y las recomendaciones top-N, donde una lista de N ítems con puntajes es retornada por el usuario. Por lo tanto, la precisión de los otros artículos, en los cuales el usuario no tendrá interés, no es importante [16]. La recomendación top-N es frecuentemente utilizada por los servicios de e-commerce donde el espacio disponible en la interfaz gráfica para listar recomendaciones es limitada y los usuarios sólo pueden ver los N ítems más evaluados. Así, MAE y todas las métricas de error en general, no son significativas para tareas de ordenamiento de ítems.

4.3.5. FCP

La Fracción de Pares Concordantes (FCP, por sus siglas en inglés) es una métrica de calidad definida por Koren et al. [22] que se utiliza para medir la proporción de ítems que fueron correctamente clasificados por el algoritmo recomendador. Esta métrica evalúa que si un usuario u para el ítem i evaluó con mayor puntaje al ítem j, es decir $r_{u,i} > r_{u,j}$ entonces esta tendencia deberá mantenerse en los resultados de las predicciones de los algoritmos recomendadores evaluados. Ésta métrica controla que tan bien un recomenda-

dor predice correctamente los intereses de los usuarios, es decir cómo el predictor ordena los puntajes de los ítems respetando el orden de los puntajes reales del usuario.

El número de pares concordantes para un usuario u calcula la cantidad de pares de calificaciones que fueron correctamente ordenadas por una predicción $\hat{r_u}$ de la siguiente manera:

$$n_c^u = |\{(i,j)|\hat{r}_{u,i} > \hat{r}_{u,j} \text{ y } r_{u,i} > r_{u,j}\}| \tag{4.5}$$

De manera similar se calculan la cantidad de pares discordantes del usuario n_d^u:

$$n_d^u = |\{(i,j)|\hat{r}_{u,i} \geq \hat{r}_{u,j} \text{ y } r_{u,i} < r_{u,j}\}| \tag{4.6}$$

Sumando sobre todos los usuarios se definen $n_c = \sum_u n_c^u$ y $n_d = \sum_u n_d^u$. Finalmente la proporción de ítems calificados correctamente se obtiene de la siguiente manera:

$$FCP = \frac{n_c}{n_c + n_d} \tag{4.7}$$

4.4. Comparación y discusión de métricas

En la sección 4.3 se definieron métricas de error y la métrica de calidad FCP para evaluar los sistemas recomendadores. En ésta sección se compararán los valores de las métricas de error más utilizadas, MAE y RMSE, y se constratarán con la métrica de calidad FCP para poder comprender cómo se calculan y cómo se evalúan.

Tanto MAE como RMSE calculan el error promedio de las predicciones de un algoritmo. Ambas métricas tienen un rango de valores de 0 a ∞. MAE, como se observa en la ecuación 4.3, calcula el valor absoluto de la diferencia entre el valor real y predicho mientras que RMSE, como se observa en la ecuación 4.2, eleva la diferencia al cuadrado, por lo que son indiferentes al signo de la diferencia entre la predicción y el valor real.

Ejemplo 1: Sea el puntaje real $r = 6$ y la predicción $\hat{r} = 4$ con un error de predicción de 2 puntos menos, MAE calculará el error de predicción para estos valores:

$$|4 - 6| = | - 2| = 2$$

mientras que para RMSE el error de predicción se calculará:

$$(4 - 6)^2 = (-2)^2 = 4.$$

En cambio si la predicción hubiera sido $\hat{r} = 8$, es decir que el predictor obtuvo un puntaje con 2 valores más grande al puntaje al real, MAE se calculará:

$$|8 - 6| = |2| = 2$$

y RMSE será:

$$(8 - 6)^2 = (2)^2 = 4,$$

por lo que se obtendrá el mismo valor para cada una de las métricas siendo indiferente si la predicción era más grande o más chica que el valor real.

Una diferencia entre MAE y RMSE es que en ésta última los errores son elevados al cuadrado antes de ser promediados, ésta métrica da pesos relativamente altos a errores muy grandes. Esto significa que RMSE será más útil cuando los errores grandes no son particularmente deseados. Mientras que en MAE todas las diferencias individualmente tienen el mismo peso.

Ejemplo 2: Sea un usuario u cuyas predicciones reales para un conjunto de 4 ítems se conocen y un algoritmo de recomendación obtuvo los puntajes para esos ítems. La siguiente tabla de ejemplo muestra para cada ítem los errores de predicción:

item	$error$	$\|error\|$	$error^2$
1	2	2	4
2	2	2	4
3	4	4	16
4	8	8	64

Tabla 4.1: Ejemplo para calcular MAE y RMSE

Si se calcula MAE y RMSE para los errores de la tabla:

$$MAE = \frac{1}{4}(2+2+4+8) = \frac{16}{4} = 4$$
$$RMSE = \sqrt{\frac{1}{4}(4+4+16+64)} = \sqrt{\frac{88}{4}} = 4,69$$

Se puede observar que RMSE da un valor superior a MAE porque RMSE le da un peso superior a los errores más grandes.

Ejemplo 3: Dada la siguiente tabla de ejemplo de prueba 4.2 con puntajes otorgados por usuarios a ítems y las predicciones obtenidas con un recomendador se mostrará como calcular FCP.

usuario	ítem	calificación	predicción	$\|error\|$	$error^2$
1	1	2	4	2	4
1	2	4	5	1	1
2	3	1	3	2	4
2	4	3	5	2	4

Tabla 4.2: Ejemplo de matriz de puntajes para evaluar la métrica FCP

Para poder calcular la métrica primero se debe contabilizar los pares concordantes n_c^u y discordantes n_d^u para cada uno de los usuarios.

El usuario 1 evaluó a los ítems 1 y 2 por lo que se deben evaluar los pares $(1, 2)$ y $(2, 1)$.

- par $(1, 2)$:
  - ¿Es concordante? Utilizando la fórmula 4.5

    $$¿(2 > 4) \text{ y } (4 > 5) \text{ ? No} \rightarrow n_c^1 = 0$$

  - ¿Es disconcordante? Utilizando la fórmula 4.6

    $$¿(2 < 4) \text{ y } (4 \geq 5) \text{ ? No} \rightarrow n_d^1 = 0$$

- par $(2, 1)$:
  - ¿Es concordante? Utilizando la fórmula 4.5

    $$¿(4 > 2) \text{ y } (5 > 4) \text{ ? Si} \rightarrow n_c^1 = 1$$

  - ¿Es disconcordante? Utilizando la fórmula 4.6

    $$¿\ (4 < 2) \text{ y } (5 \geq 4) \text{ ? No} \rightarrow n_d^1 = 0$$

Entonces para el usuario 1 la cantidad de pares concordantes y discordantes es la siguiente:

$$n_c^1 = 1 \text{ y } n_d^1 = 0$$

Por otro lado, el usuario 2 evaluó a los ítems 3 y 4 por lo que se deben evaluar los pares $(3, 4)$ y $(4, 3)$.

- par $(3, 4)$:
  - ¿Es concordante? Utilizando la fórmula 4.5

    $$¿(1 > 3) \text{ y } (3 > 5) \text{ ? No} \rightarrow n_c^2 = 0$$

  - ¿Es disconcordante? Utilizando la fórmula 4.6

    $$¿(1 < 3) \text{ y } (3 \geq 5) \text{ ? No} \rightarrow n_d^2 = 0$$

- par $(4, 3)$:
  - ¿Es concordante? Utilizando la fórmula 4.5

    $$¿(3 > 1) \text{ y } (5 > 3) \text{ ? Si} \rightarrow n_c^2 = 1$$

  - ¿Es disconcordante? Utilizando la fórmula 4.6

    $$¿(3 < 1) \text{ y } (5 \geq 3) \text{ ? No} \rightarrow n_d^2 = 0$$

Entonces para el usuario 2 la cantidad de pares disconcordantes y concordantes es:

$$n_c^2 = 1 \text{ y } n_d^2 = 0$$

Sumando para todos los usuarios:

$$n^c = n_c^1 + n_c^2 = 1 + 1 = 2 \text{ y } n^d = n_d^1 + n_d^2 = 0$$

Finalmente, FCP se calcula de acuerdo a la ecuación 4.7 de la siguiente manera:

$$FCP = \frac{2}{2+0} = 1$$

El máximo valor posible que puede devolver esta métrica de calidad es 1. Esto quiere decir que el predictor mantiene el orden de todas las calificaciones según los intereses de los usuarios. Sin embargo, se puede observar en la tabla 4.2 que las predicciones para cada uno de los ítems no fueron precisas ya que en la mayoría de los casos el algoritmo predice con una diferencia de 2 puntos con respecto al puntaje real.

Este error de predicción se puede observar calculando las métricas MAE y RMSE de acuerdo a las ecuaciones 4.3 y 4.2.

$$MAE = \frac{1}{4}(2+1+2+2) = \frac{7}{4} = 1,75$$

$$RMSE = \sqrt{\frac{1}{4}(4+1+4+4)} = \sqrt{\frac{13}{4}} = 1,802$$

Así también se observa que como la varianza de la distribución de los errores es pequeña RMSE da un valor muy cercano a MAE. En este ejemplo se muestra que si un recomendador tiene un FCP muy alto no significa que no tenga errores en las predicciones.

Qué métrica interesará observar dependerá de cual sea nuestro interés respecto a los errores de predicción. Si interesa minimizar los errores de predicción, alcanzaría con evaluar MAE. Si interesa que no se obtengan errores de predicción muy grandes se debería utilizar RMSE. Y si interesa que se mantenga un orden en las calificaciones de los usuarios, sin que importe tanto la predicción exacta de la calificación, se debería utilizar FCP. Por otro lado, cabe destacar que éstas métricas se utilizan para evaluar y comparar diferentes recomendadores y elegir el que mejor se ajuste de acuerdo a las métricas que se interesen observar.

4.5. Evaluación de los Sistemas Recomendadores aplicados en Educación

La evaluación de los SR aplicados en Educación se lleva a cabo principalmente en forma de experimentos offline como así también a través de la realización de estudios de usuarios controlados. En lo que respecta a los experimentos offline, en su mayoría siguen el enfoque típico de pruebas mostradas en las secciones anteriores. Drachshler et al. [1], Sicilia et al. [52] y Verbert et al. [36] adoptan y muestran estos enfoques utilizando conjuntos de datos de aplicaciones y entornos educativos. Por su lado, los estudios de usuarios controlados son muy valiosos y muy utilizados en entornos educativos [1]. Este tipo de experimentos también pueden contribuir a evaluar aspectos tecnológicos específicos del sistema, además de las evidencias empíricas de los aspectos psicológicos y pedagógicos que pueden recopilarse mediante estos estudios de usuarios controlados.

En educación para evaluar los SR impulsados por la pedagogía tanto para el aprendizaje formal y no formal, no hay disponibles conjuntos de datos ni procedimientos de evaluación estandarizados. Además, centrarse únicamente en las métricas de evaluación de los SR sin tener en cuenta las necesidades reales y las características de los alumnos y los docentes es cuestionable. Por lo tanto, se necesitan más procedimientos de evaluación que complementen los enfoques de evaluación técnica. Por ejemplo, los alumnos solo se benefician de los sistemas compatibles y mejorados de AMT cuando hacen que el aprendizaje sea más efectivo, eficiente y / o más atractivo [3] [68].

Según Manouselis et al. [3], las medidas comunes para evaluar el éxito de tales sistemas en entornos educativos incluyen: la eficacia, la eficiencia, la satisfacción y la tasa de abandono. La efectividad es un signo de la cantidad total de contenidos completados, visitados o estudiados durante una fase de aprendizaje. La eficiencia indica el tiempo que los estudiantes necesitan para alcanzar su meta de aprendizaje, que se relaciona con la variable de efectividad contando el tiempo de estudio real. La satisfacción refleja la

satisfacción individual de los alumnos con las recomendaciones dadas, y ésta se puede relacionar con la motivación de un alumno y, por lo tanto, es una medida importante para el aprendizaje. Mientras que la tasa de abandono refleja el número de estudiantes que abandonan durante la fase de aprendizaje.

En lo que respecta a las redes de aprendizaje, los métodos y métricas que se originan en el Análisis de redes sociales (SNA) [58], también se podrían contrastar para medir el éxito de los recomendadores de AMT. SNA ofrece varias ideas sobre los diferentes roles que los alumnos pueden tener en una red de aprendizaje. Las medidas típicas de SNA son variedad, centralidad, proximidad y cohesión. La variedad mide el nivel de emergencia en una red de aprendizaje a través de la combinación de caminos de aprendizaje individuales con las rutas de aprendizaje más exitosas. La centralidad es un indicador de la conectividad de un alumno en una red de aprendizaje. Cuenta el número de vínculos con otros estudiantes en la red. La cercanía mide el grado en que un alumno está cerca de todos los demás alumnos en una red. Representa la capacidad de acceder a la información directa o indirectamente a través de la conexión a otros miembros de la red. La cohesión, por otro lado, indica qué tan fuertemente los estudiantes están directamente conectados entre sí por enlaces cohesivos. Se pueden identificar grupos de pares de alumnos si cada alumno está directamente vinculado a cada uno de los otros aprendices en la red de aprendizaje. Drachsler et al. [1] siguieron este enfoque mediante el uso de simulaciones para evaluar el impacto de un SR en las redes de aprendizaje informal.

Sintetizar todos los componentes en un marco de evaluación general tiene varias dificultades metodológicas y prácticas. Como una guía general, sin embargo, los marcos clásicos de evaluación de la comunidad educativa podrían adoptarse y ser adaptado al contexto de los SR. El modelo de Kirckpatrick [69] mide el éxito de una prueba utilizando cuatro capas de evaluación diferentes, y esto podría utilizarse para evaluar el éxito de un SR en un contexto AMT:

- Reacción del usuario: lo que ellos pensaron y sintieron (“¿Disfruté las recomenda-

ciones que recibí?");

- Aprendizaje: el aumento resultante en la obtención de nuevos conocimientos o capacidades
  ("¿Aprendí lo que necesitaba y obtuve algunas ideas nuevas con la ayuda del recomendador?");

- Comportamiento: medida de cómo se puede implementar/aplicar el conocimiento y la capacidad adquirida en la vida real
  ("¿Usaré la información e ideas nuevas que me recomendaron?");

- Resultados: los efectos sobre el rendimiento del usuario en el entorno de aprendizaje o trabajo
  ("¿Las ideas y la información que me recomendaron mejoran mi efectividad y resultados?").

Por lo tanto, la definición de un marco de evaluación general en el ámbito educativo podría incluir:

- Un análisis detallado de los métodos de evaluación y las herramientas que se pueden emplear para evaluar las técnicas de recomendación en educación contra un conjunto de criterios que se definirán para cada uno de los componentes seleccionados (por ejemplo, modelo de usuario, modelo de dominio, estrategia de recomendación y algoritmo). Para el ejemplo presentado de las dimensiones de Kirckpatrick, esto incluiría una identificación de los métodos de evaluación que podrían emplearse para medir el efecto del recomendador en un contexto de enseñanza y aprendizaje en particular.

- La especificación de métricas, indicadores de evaluación para medir el éxito de cada componente (por ejemplo, evaluar la precisión del algoritmo de recomendación, evaluar la cobertura del modelo de dominio).

- La elaboración de una serie de métodos e instrumentos que pueden emplearse con el fin de recopilar datos de evaluación de partes interesadas involucradas, explícita o implícitamente, para medir la satisfacción tanto de docentes y alumnos, y para poder evaluar el impacto de incorporar un recomendador en el ámbito educativo.

Capítulo 5

Trabajo Experimental

5.1. Configuración del experimento

En el ámbito de los Sistemas Recomendadores aplicados al comercio electrónico distintos conjuntos de datos con características específicas están disponibles, como por ejemplo el de MovieLens de películas, el de Book-Crossing de libros, y el de Jester de música. Estos datasets son usados comúnmente como *benchmark* para evaluar nuevos algoritmos de recomendación. En el área educativa los datos de prueba que generalmente se utilizan son los que participaron del primer dataTEL challenge realizado en 2010 [59], donde estos datos educativos capturaban interacciones de estudiantes con herramientas y recursos educativos. Es importante destacar que mucho de los datasets pertenecían a proyectos que ya han finalizado y muchos de ellos no se encuentran fácilmente o no están disponibles públicamente para su descarga.

Para evaluar los distintos algoritmos de recomendación se consiguieron tres datasets: el de MACE [60] y el de Travel Well [61] que formaron parte del dataTEL challenge mencionado anteriormente, y el de Open University recolectado por Kuzilek et al. [70] que se describen a continuación:

- **MACE**

  Este dataset pertenece al proyecto MACE que estuvo en servicio desde septiembre de 2006 hasta septiembre de 2009. En el proyecto se creó una plataforma virtual de aprendizaje informal que vinculaba diferentes repositorios de toda Europa para proporcionar acceso a los recursos de aprendizaje a través de metadatos del dominio. Los estudiantes podían buscar, agregar competencias, agregar etiquetas descriptivas, mirar los metadatos, mirar el contenido y evaluar los recursos que fueran apropiados para su contexto. En su período de actividad, 1148 usuarios se registraron en el portal, donde se ofrecían alrededor de 150.000 recursos, y donde 12.500 de éstos llegaron a ser accedidos por usuarios registrados. El dataset dispone información de 628 usuarios, 12.367 recursos de aprendizaje y 117.878 interacciones entre usuarios e ítems. Las interacciones se dividen en: addCompetence (si el usuario agregó una competencia), getMetadataForContent (si el usuario accedió a la metadata del recurso), goToPage (si accedió al contenido del recurso), addTag (si el usuario agregó una etiqueta descriptiva al recurso) y rate (si el usuario evaluó o no a un recurso). En la figura 5.1 se observa la distribución de eventos, se observa que el evento goToPage representa casi un 8 % y el evento rate no llega al 1 %, por lo que casi no se disponen de calificaciones.

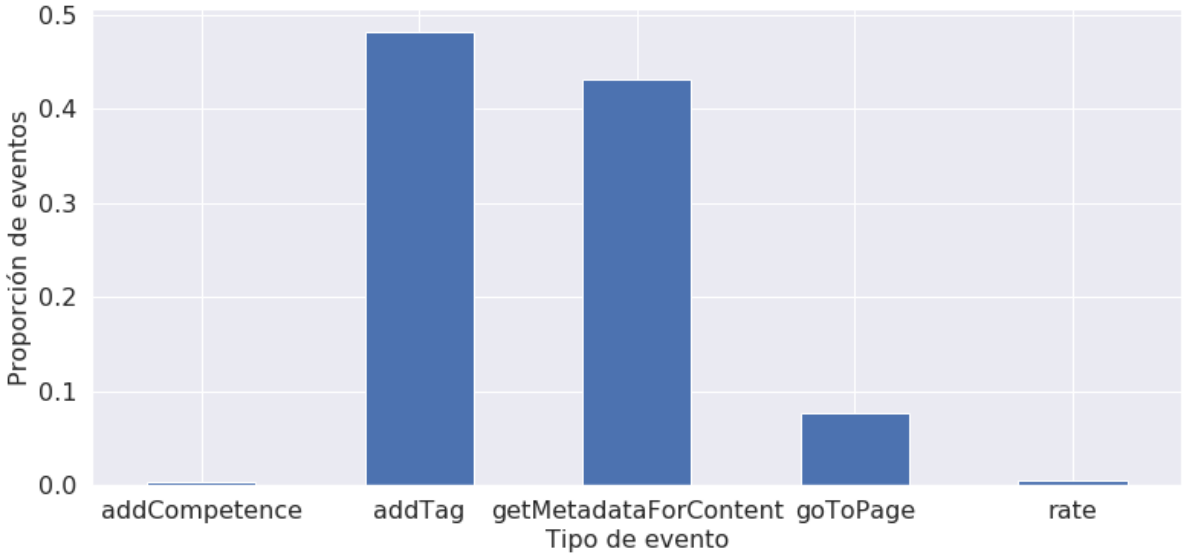


Figura 5.1: Dataset de MACE. Distribución de los eventos

- **Travel Well**

  Este dataset se recopiló del portal Travel Well que recomendaba Recursos Educativos Abiertos de 20 proveedores de contenido en Europa y otros lugares. La mayoría de los usuarios registrados eran profesores de primaria y secundaria de países europeos. Este conjunto de datos contiene datos de la fase piloto que se llevó a cabo durante el proyecto MELT. Estos datos se recopilaron desde agosto de 2008 hasta febrero de 2009 con 98 usuarios. Los usuarios podían calificar los recursos en una escala de 1 a 5 y agregar etiquetas a los recursos. En total, se registraron 16.353 actividades de usuarios en 1.923 recursos. El conjunto de datos contiene información de 75 usuarios, 1.608 ítems y 2.156 calificaciones. Además de los usuarios se conoce el país de origen, la lengua materna y los idiomas hablados. Además, de cada ítem se tienen metadatos sobre el origen del mismo y su idioma. En la figura 5.2 se observa la distribución de calificaciones para este dataset.

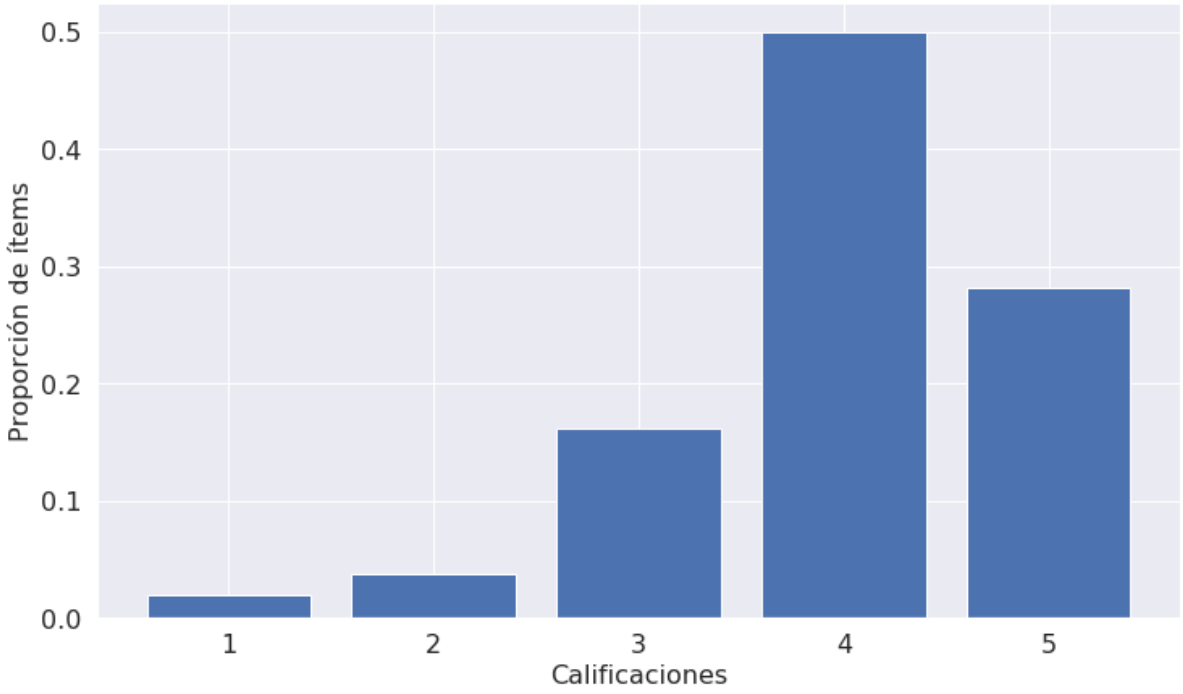


Figura 5.2: Dataset de Travel Well. Distribución de calificaciones

- **Open University Learning Analytics**

  Este dataset contiene información de alumnos de la Open University (OU), una universidad pública que se encuentra en Reino Unido y que brinda cursos a distancia. Kuzilek et al. [70] recolectaron del EVEA de la OU información entre los años 2013 y 2014 sobre 22 cursos, 26.074 alumnos, resultados de evaluaciones y las interacciones con 6.268 recursos educativos a través de un resumen diario de los clicks realizados por los alumnos (10.655.280 interacciones). A su vez, de los alumnos se conoce su edad, sexo, de que región son, último nivel de educación obtenido, en qué cursos están inscriptos, cantidad de veces que el alumno se anotó a un mismo curso, si aprobaron o no, y si posee una capacidad diferente. Mientras que de cada recurso además se conoce su tipo de actividad (recurso, página web, etc). Este dataset se

	MACE	**Travel Well**	**Open University**
# Usuarios	628	75	26,074
# Ítems	12,367	1,608	6,268
# Interacciones	117,878	2,156	10,655,280
Calificaciones	Si (si realizó el evento rate)	Si	No
Eventos	addCompetence, getMetadataForContent, goToPage, addTag, rate	No	sum_clicks

Tabla 5.1: Resumen de los datasets a utilizar

encuentra disponible bajo licencia Creative Commons BY 4.0.

En la tabla 5.1 se muestra un resumen de los datos de prueba que se utilizarán.

Los algoritmos presentados en la sección 2.3 requieren calificaciones numéricas para poder obtener las recomendaciones y con el objetivo de poder comparar estos diferentes enfoques se procesarán los datasets que tienen eventos para transformarlos en una misma valoración numérica.

Para los datos de MACE se tendrán en cuenta los eventos “rate” o “goToPage”, quedando un dataset de 9629 eventos, 566 alumnos y 5053 ítems. Si el usuario realizó el evento “rate” se pondrá como puntaje esa misma valoración, el rango de puntajes es de 1 al 10, y si el usuario no realizó este evento se considerará el evento “goToPage” que significa que el usuario abrió el recurso. Si el usuario realizó este evento más de dos

veces se pondrá calificación 4 y si en cambio lo vio una sola vez el puntaje será 2. Para el dataset de Open University se dispone la suma de clicks que los usuarios realizaron sobre cada uno de los ítems que se transformarán en puntajes del 1 al 3. Si el usuario hizo más de 3 clicks sobre un recurso el puntaje será 3, si realizó 2 clicks el puntaje será 2, y si realizó un solo click el puntaje será 1.

Se probarán las siguientes técnicas de recomendación: Filtrado Colaborativo basado en el enfoque de vecinos más próximos usuario-usuario e ítem-ítem, utilizando el algoritmo que utiliza un baseline y la variante que utiliza el puntaje promedio ambos descriptos en la sección 2.3.1; y por otro lado se probará también el enfoque de factorización de matrices SVD descripto en la sección 2.3.2.

La metodología de evaluación para los algoritmos a utilizar aplica validación cruzada 10-fold. Este proceso de evaluación se repitió 50 veces para obtener una muestra significativa sobre la cual se promedian los resultados. Los métodos kNN reciben como parámetro la cantidad de vecinos a considerar donde el tamaño del vecindario tiene un impacto significativo en la calidad de la predicción [16], por lo que previamente se deberá calcular el tamaño k de vecinos óptimo para cada uno de los datos de prueba. Finalmente, para evaluar las predicciones de los algoritmos de recomendación elegidos se calcularon las métricas de error RMSE y MAE y la métrica de calidad FCP que mide la proporción de pares de ítems bien clasificados vistas en la sección 4.3.

5.2. Resultados

Para encontrar la cantidad de vecinos para cada uno de los datasets de prueba se calculó el error RMSE visto en la sección 4.3 para diferentes números de vecinos en los distintos algoritmos, donde el error decrece a medida que la cantidad de vecinos crece. Se elige el número de vecinos óptimo a partir del k donde el error empieza a decrecer lentamente [16].

Para el dataset de MACE se utilizará un tamaño de vecindad $k = 10$, para Travel Well $k = 15$ y para Open University $k = 20$. En la figura 5.3 se puede observar como el error decrece lentamente a partir del k elegido para todos los algoritmos y además se observa como este error da menor valor para el algoritmo basado en el enfoque kNN Baseline ítem-ítem.

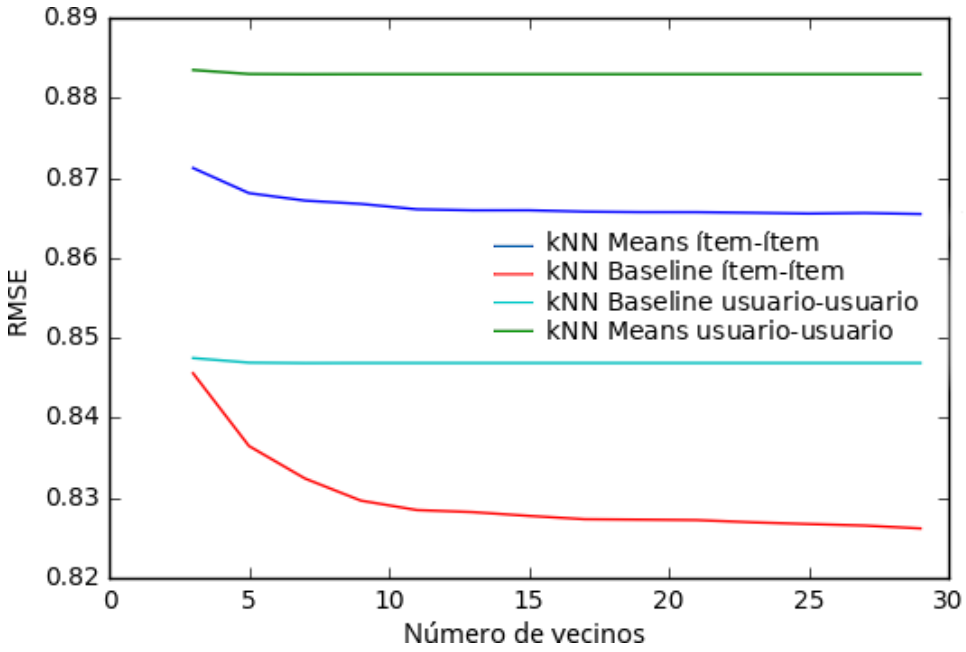


Figura 5.3: Dataset de Travel Well. Influencia del tamaño de vecindad

Los resultados de las 50 ejecuciones de la validación cruzada para cada algoritmo y cada métrica utilizando los diferentes datasets educativos se muestran en la tabla 5.2. De cada métrica se observa los resultados y las desviaciones estándar obtenidas, los mejores resultados están resaltados en negrita. A diferencia de las métricas de error RMSE y MAE, el valor de FCP es mejor cuanto más alto es, porque mide una proporción.

Tabla 5.2: Resultados obtenidos para los distintos conjuntos de datos

Algoritmo	**Métrica**	**Dataset**		
		MACE	Travel Well	Open University
kNN Baseline ítem-ítem	FCP	0,896 ± 0,013	0,451 ± 0,16	0,551 ± 0,002
	MAE	0,465 ± 0,008	0,601 ± 0,094	0,772 ± 0,002
	RMSE	0,623 ± 0,01	0,814 ± 0,122	0,915 ± 0,002
kNN Baseline usuario-usuario	FCP	0,919 ± 0,026	0,463 ± 0,143	0,544 ± 0,003
	MAE	**0,121 ± 0,006**	0,59 ± 0,105	0,743 ± 0,003
	RMSE	0,283 ± 0,018	0,824 ± 0,142	0,908 ± 0,004
kNN Means ítem-ítem	FCP	0,905 ± 0,008	**0,47 ± 0,152**	**0,564 ± 0,003**
	MAE	0,303 ± 0,008	**0,555 ± 0,105**	**0,733 ± 0,001**
	RMSE	0,499 ± 0,015	0,843 ± 0,138	0,898 ± 0,002
kNN Means usuario-usuario	FCP	0,922 ± 0,021	0,445 ± 0,152	0,551 ± 0,004
	MAE	0,136 ± 0,006	0,582 ± 0,107	0,765 ± 0,003
	RMSE	0,301 ± 0,018	0,86 ± 0,14	0,944 ± 0,004
SVD	FCP	**0,944 ± 0,031**	0,46 ± 0,136	0,556 ± 0,002
	MAE	0,148 ± 0,004	0,603 ± 0,098	0,741 ± 0,002
	RMSE	**0,269 ± 0,012**	**0,808 ± 0,129**	**0,839 ± 0,002**

Para el dataset Travel Well, el algoritmo SVD obtuvo un menor error RMSE seguido por el método kNN Baseline ítem-ítem. El error MAE dió un valor menor para kNN Means ítem-ítem seguido por kNN Means usuario-usuario. Mientras que la métrica FCP fue más alta para kNN Means ítem-ítem seguido por kNN Baseline usuario-usuario.

En el dataset MACE, el algoritmo SVD tiene menor error RMSE y luego le sigue el método kNN Baseline usuario-usuario. Para el error MAE los enfoques usuario-usuario arrojan valores muy similares. Mientras que la métrica FCP tiene mejor resultado para SVD seguido por kNN Baseline usuario-usuario con un valor similar.

Por último para el dataset de Open University, el error RMSE es mejor para SVD y luego le sigue el algoritmo kNN Means ítem-ítem. Con el error MAE, kNN Means ítem-ítem ofreció mejores resultados seguido por SVD. Mientras que kNN Means ítem-ítem arrojó mejores resultados en la métrica FCP seguido por SVD.

Es posible observar que el algoritmo SVD obtiene mejores resultados para la métrica RMSE, esto se debe a que el algoritmo intenta justamente minimizar este error a la hora de generar el modelo de predicción. Es de esperarse que presente mejores resultados para esta métrica.

Los enfoques ítem-ítem y las técnicas de descomposición de matrices otorgan mejores resultados en las pruebas llevadas a cabo. Estos resultados son también de esperarse porque los enfoques ítem-ítem tienen una mejor escalabilidad y precisión que los enfoques usuario-usuario.

Los enfoques usuario-usuario arrojaron mejores resultados para MACE. Este dataset originalmente está basado en eventos y no principalmente por calificaciones. La cantidad de eventos promedio por usuarios de este dataset es de 17 eventos por usuario, mientras que la cantidad de eventos promedio por ítem es aproximadamente 2 eventos por ítem. En la figura 5.4 se muestra la cantidad de eventos realizados por los usuarios, y se puede observar que más de la mitad de los usuarios tienen más de 5 eventos realizados, con una mediana de 7 eventos por usuario. Por lo que se dispone de más información a nivel

usuarios con respecto a la información a nivel de los ítems. Por esta razón los enfoques basados en usuario-usuario otorgan mejores resultados en este contexto.

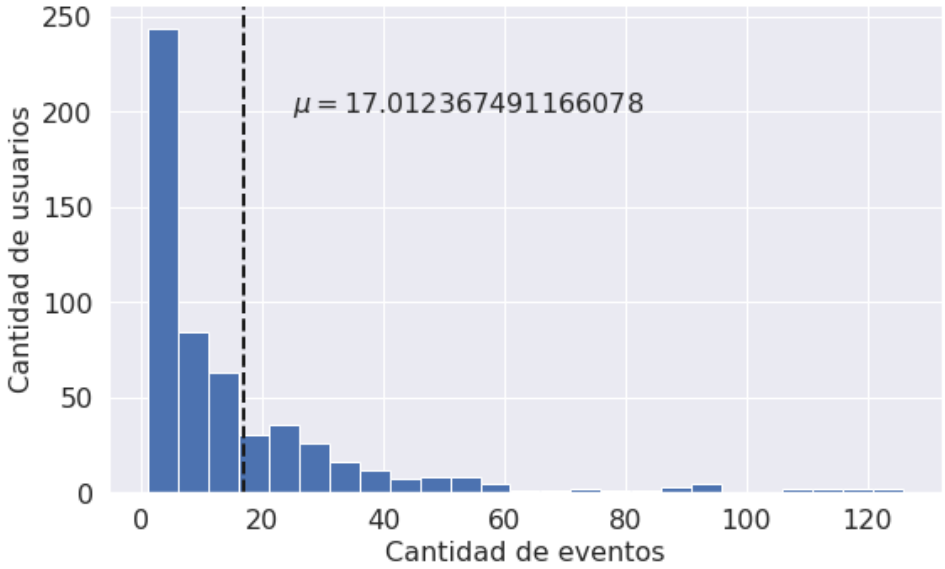


Figura 5.4: Dataset de MACE. Cantidad de eventos realizados por los usuarios

Capítulo 6

Conclusiones y líneas de trabajo futuro

En este trabajo final integrador se analizaron diferentes técnicas de recomendación y se estudió su aplicabilidad en el ámbito educativo. Así también se presenta un resumen de las métricas usualmente utilizadas para medir la performance de éstos sistemas y cuáles son las variantes o nuevas métricas a tener en cuenta cuando se aplican éstos sistemas en educación. En el trabajo experimental se utilizaron diferentes conjuntos de datos de prueba abordados en la literatura de los SRE y se compararon los resultados obtenidos con distintos algoritmos de recomendación basados en la técnica de FC.

El presente trabajo se enmarca en una línea de investigación cuyo punto central trata sobre el estudio, diseño y desarrollo de nuevas técnicas adaptativas, pertenecientes al área de la Analítica del Aprendizaje, que contribuyan en la toma de decisiones. El énfasis está puesto en la construcción de un SR inteligente con capacidad para asistir en ámbitos educativos.

Los SR aplicados en Educación permiten que los alumnos puedan encontrar materiales que se ajusten a sus necesidades y preferencias, y que los materiales recomendados se adapten a los objetivos pedagógicos de los docentes. Para poder mejorar la calidad de las

recomendaciones, es fundamental contar con la mayor cantidad de información posible para poder modelar a los usuarios que intervienen y a los contenidos.

Dentro de esta línea de investigación, inicialmente se estudió como modelar los perfiles de los alumnos de la Facultad de Informática de la UNLP. Por medio de técnicas de aprendizaje automático, procesamiento de lenguaje natural y técnicas de visualización de datos masivos se construyeron modelos para caracterizar a los alumnos y descubrir factores latentes implícitos utilizando su información académica y sus intereses e interacciones en las redes sociales [71] [72] [15].

A partir del trabajo experimental realizado en el presente trabajo se pudo mostrar como los enfoques de Filtrado Colaborativo ofrecen buenos resultados utilizando materiales educativos valorados por los usuarios. Las técnicas basadas en el enfoque ítem-ítem de modelos de vecindad son uno de los algoritmos de recomendación más utilizados por su simpleza y eficacia, los cuales se pueden adaptar adecuadamente dentro de un ámbito educativo. Sin embargo, es importante tener en cuenta que los experimentos presentados sirven solo como un primer paso hacia la comprensión y la especialización apropiada para la recomendación en este dominio [36]. Este estudio debe complementarse con experimentos que tengan en cuenta las necesidades y expectativas de los usuarios, sus tareas de búsqueda de información y cómo los recursos recomendados pueden usarse en el contexto de sus actividades de aprendizaje [36] [3]. Además, algunas consideraciones que se pueden tener en cuenta al aplicar un algoritmo de recomendación en el ámbito educativo puede implicar desde qué métricas de evaluación se deben utilizar hasta cómo se puede complementar el recomendador por medio de información provista por los docentes y/o a partir de información adicional sobre los recursos educativos a sugerir para asegurar una mejor experiencia a los alumnos [3] [12].

Generalmente, los enfoques híbridos complementan un algoritmo de FC con otro tipo de técnicas y/o estrategias para poder potenciar sus recomendaciones de acuerdo al dominio donde se lo quiera aplicar. Dentro del marco de investigación antes mencionado,

actualmente se está trabajando en cómo utilizar técnicas de procesamiento de lenguaje natural para poder modelar los contenidos de los recursos educativos y poder mejorar la performance de un SR. En este contexto se desarrolló un nuevo método de recomendación que descubre de forma automática los temas que tratan los recursos educativos a partir de su información textual y se estableció una nueva medida de similitud para poder compararlos [13] [73].

Finalmente, es importante destacar que como se discutía en el capítulo 3, es imprescindible contar con un mayor número de datasets de prueba de recomendadores de materiales educativos que sean de libre acceso y que cuenten con la información necesaria para poder probar nuevas variantes y métricas de recomendación en este ámbito.

Bibliografía

[1] Drachsler, H., Verbert, K., Santos, O.C., Manouselis, N.: Panorama of recommender systems to support learning. In: Recommender systems handbook. Springer (2015) 421–451

[2] Đurović, G., Holenko Dlab, M., Hoić-Božić, N.: Educational recommender systems: An overview and guidelines for further research and development. Volume 20., Učiteljski fakultet; Sveučilišta u Zagrebu (2018) 531–560

[3] Manouselis, N., Drachsler, H., Verbert, K., Duval, E.: Recommender systems for learning. Springer Science & Business Media (2012)

[4] Peralta, M., Alarcon, R., Pichara, K.E., Mery, T., Cano, F., Bozo, J.: Understanding learning resources metadata for primary and secondary education. IEEE Transactions on Learning Technologies (2017)

[5] Bonde, S.N., Kirange, D.: Educational data mining survey for predicting student's academic performance. In: International conference on Computer Networks, Big data and IoT, Springer (2018) 293–302

[6] Purwoningsih, T., Santoso, H.B., Isal, Y.K., Hasibuan, Z.A.: The pedagogy optimization with educational data mining and learning analytics for e-learning system-a review of the literature review. In: 2018 Third International Conference on Informatics and Computing (ICIC), IEEE (2018) 1–5

[7] Al-Shibly, M.S.: The use of social media in knowledge sharing case study undergraduate students in major british universities. International Journal of Online Marketing (IJOM) **9** (2019) 19–32

[8] Ricci, F., Rokach, L., Shapira, B.: Recommender systems: introduction and challenges. In: Recommender systems handbook. Springer (2015) 1–34

[9] Bansal, S., Baliyan, N.: A study of recent recommender system techniques. International Journal of Knowledge and Systems Science (IJKSS) **10** (2019) 13–41

[10] Liu, Y., Qu, H., Chen, W., Mahmud, S.H.: An efficient deep learning model to infer user demographic information from ratings. IEEE Access **7** (2019) 53125–53135

[11] Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) **22** (2004) 5–53

[12] Hosseini, R., Brusilovsky, P.: A comparative study of visual cues for annotation-based navigation support in adaptive educational hypermedia. In: CEUR Workshop Proceedings. Volume 1618. (2016)

[13] Charnelli, M.E., Lanzarini, L., Díaz, J.: Recommender system based on latent topics. In: Argentine Congress of Computer Science, Springer (2017) 179–187

[14] Charnelli, M.E., Lanzarini, L.C., Baldino, G., Díaz, F.J.: Determining the profiles of young people from buenos aires with a tendency to pursue computer science studies. In: XX Congreso Argentino de Ciencias de la Computación. Selected Papers. (2015)

[15] Charnelli, M.E., Lanzarini, L., Diaz, J.: Modeling students through analysis of social networks topics. XXII Congreso Argentino de Ciencias de la Computacion CACIC 2016 (2016) 363–371

[16] Kluver, D., Ekstrand, M.D., Konstan, J.A.: Rating-based collaborative filtering: algorithms and evaluation. In: Social Information Access. Springer (2018) 344–390

[17] Ling, Z., Xiao, Y., Wang, H., Xu, L., Hsu, C.H.: Extracting implicit friends from heterogeneous information network for social recommendation. In: Pacific Rim International Conference on Artificial Intelligence, Springer (2019) 607–620

[18] Sapountzi, A., Psannis, K.E.: Social networking data analysis tools & challenges. Future Generation Computer Systems **86** (2018) 893–913

[19] Liang, D., Krishnan, R.G., Hoffman, M.D., Jebara, T.: Variational autoencoders for collaborative filtering. In: Proceedings of the 2018 World Wide Web Conference, International World Wide Web Conferences Steering Committee (2018) 689–698

[20] Cunha, T., Soares, C., de Carvalho, A.C.: Metalearning and recommender systems: A literature review and empirical study on the algorithm selection problem for collaborative filtering. Information Sciences **423** (2018) 128–144

[21] Kharita, M.K., Kumar, A., Singh, P.: Item-based collaborative filtering in movie recommendation in real time. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), IEEE (2018) 340–342

[22] Koren, Y., Sill, J.: Collaborative filtering on ordinal user feedback. In: IJCAI. (2013) 3022–3026

[23] Gunawardana, A., Shani, G.: Evaluating recommender systems. In: Recommender systems handbook. Springer (2015) 265–308

[24] Rojas, L.F.L., Marín, P.A.R., Méndez, N.D.D.: Comparative analysis of similarity metrics for the collaborative recommendation of learning objects. In: Learning Technologies (LACLO), 2017 Twelfth Latin American Conference on, IEEE (2017) 1–4

[25] Melville, P., Sindhwani, V.: Recommender systems. Encyclopedia of Machine Learning and Data Mining (2017) 1056–1066

[26] Luo, X., Wang, D., Zhou, M., Yuan, H.: Latent factor-based recommenders relying on extended stochastic gradient descent algorithms. IEEE Transactions on Systems, Man, and Cybernetics: Systems (2019)

[27] Bell, R., Koren, Y., Volinsky, C.: Modeling relationships at multiple scales to improve accuracy of large recommender systems. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2007) 95–104

[28] Funk, S.: Netflix update: Try this at home (december 2006). URL http://sifter.org/˜ simon/journal/20061211. html (2006)

[29] Imran, H., Belghis-Zadeh, M., Chang, T.W., Graf, S., et al.: Plors: a personalized learning object recommender system. Vietnam Journal of Computer Science **3** (2016) 3–13

[30] Tarus, J.K., Niu, Z., Mustafa, G.: Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. Artificial Intelligence Review **50** (2018) 21–48

[31] Sergis, S., Zervas, P., Sampson, D.G.: Towards learning object recommendations based on teachers' ict competence profiles. In: Advanced Learning Technologies (ICALT), 2014 IEEE 14th International Conference on, IEEE (2014) 534–538

[32] Fazeli, S., Drachsler, H., Brouns, F., Sloep, P.: Towards a social trust-aware recommender for teachers. In: Recommender systems for technology enhanced learning. Springer (2014) 177–194

[33] Kopeinik, S., Kowald, D., Lex, E.: Which algorithms suit which learning environments? a comparative study of recommender systems in tel. In: European Conference on Technology Enhanced Learning, Springer (2016) 124–138

[34] Manuel, S.: Estilos de Aprendizaje y Métodos de enseñanza. Tesis de Maestría. UNED (2018)

[35] Boticario, J.G., Rodriguez-Ascaso, A., Santos, O.C., Raffenne, E., Montandon, L., Roldán Martínez, D., Buendía García, F.: Accessible lifelong learning at higher education: outcomes and lessons learned at two different pilotsites in the eu4all project. In: Journal of Universal Computer Science. Volume 18., Graz University of Technology, Institut f r Informationssysteme und Computer Medien (IICM) (2012) 62–85

[36] El Guabassi, I., Al Achhab, M., Jellouli, I., Mohajir, B.E.E.: Context-aware recommender systems for learning. International Journal of Information Science and Technology **1** (2018) 17–25

[37] Harrathi, M., Touzani, N., Braham, R.: Toward a personalized recommender system for learning activities in the context of moocs. In: International Conference on Intelligent Interactive Multimedia Systems and Services, Springer (2018) 575–583

[38] Lang, C., Siemens, G., Wise, A., Gasevic, D.: Handbook of learning analytics. SOLAR, Society for Learning Analytics and Research (2017)

[39] Benhamdi, S., Babouri, A., Chiky, R.: Personalized recommender system for e-learning environment. Education and Information Technologies **22** (2017) 1455–1477

[40] Mehta, P., Saroha, K.: Recommendation system for learning management system. In: Information and Communication Technology for Sustainable Development. Springer (2018) 365–374

[41] Terveen, L., Hill, W., Amento, B., McDonald, D., Creter, J.: Phoaks: A system for sharing recommendations. Communications of the ACM **40** (1997) 59–62

[42] Chislenko, A.: Collaborative information filtering and semantic transports. WWW publication: http://www. lucifer. com/˜ sasha/articles/ACF. html (1998)

[43] Walker, A., Recker, M.M., Lawless, K., Wiley, D.: Collaborative information filtering: A review and an educational application. International Journal of Artificial Intelligence in Education **14** (2004) 3–28

[44] Dron, J., Mitchell, R., Siviter, P., Boyne, C.: Cofind—an experiment in n-dimensional collaborative filtering. Journal of Network and Computer Applications **23** (2000) 131–142

[45] Lemire, D.: Scale and translation invariant collaborative filtering systems. Information Retrieval **8** (2005) 129–150

[46] Rafaeli, S., Dan-Gur, Y., Barak, M.: Social recommender systems: recommendations in support of e-learning. International Journal of Distance Education Technologies **3** (2005) 30

[47] Avancini, H., Straccia, U.: User recommendation for collaborative and personalised digital archives. International Journal of Web Based Communities **1** (2005) 163–175

[48] Lagoze, C., Van de Sompel, H.: The open archives initiative: Building a low-barrier interoperability framework. In: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, ACM (2001) 54–62

[49] Shen, L.p., Shen, R.m.: Learning content recommendation service based-on simple sequencing specification. In: International Conference on Web-Based Learning, Springer (2004) 363–370

[50] Tang, T.Y., Winoto, P., McCalla, G.: Further thoughts on context-aware paper recommendations for education. In: Recommender Systems for Technology Enhanced Learning. Springer (2014) 159–173

[51] Nadolski, R.J., Van den Berg, B., Berlanga, A.J., Drachsler, H., Hummel, H.G., Koper, R., Sloep, P.B.: Simulating light-weight personalised recommender systems in learning networks: A case for pedagogy-oriented and rating-based hybrid recommendation strategies. Journal of Artificial Societies and Social Simulation **12** (2009) 4

[52] Sicilia, M.Á., García-Barriocanal, E., Sánchez-Alonso, S., Cechinel, C.: Exploring user-based recommender results in large learning object repositories: the case of merlot. Procedia Computer Science **1** (2010) 2859–2864

[53] Moncada, S.M.: Rediscovering merlot: A resource sharing cooperative for accounting education. Journal of Higher Education Theory & Practice **15** (2015)

[54] Salehi, M.: Application of implicit and explicit attribute based collaborative filtering and bide for learning resource recommendation. Data & Knowledge Engineering **87** (2013) 130–145

[55] Fazeli, S., Loni, B., Drachsler, H., Sloep, P.: Which recommender system can best fit social learning platforms? In: European Conference on Technology Enhanced Learning, Springer (2014) 84–97

[56] Karampiperis, P., Koukourikos, A., Stoitsis, G.: Collaborative filtering recommendation of educational content in social environments utilizing sentiment analysis techniques. In: Recommender Systems for Technology Enhanced Learning. Springer (2014) 3–23

[57] Li, H., Shi, J., Zhang, S., Yun, H.: Implementation of intelligent recommendation system for learning resources. In: Computer Science and Education (ICCSE), 2017 12th International Conference on, IEEE (2017) 139–144

[58] Kim, J., Hastak, M.: Social network analysis: Characteristics of online social networks after a disaster. International Journal of Information Management **38** (2018) 86–96

[59] 4th ACM Conference on Recommender Systems (RecSys 2010), t.E.C.o.T.E.L.E.T..: DataTEL Challenge. `http://adenu.ia.uned.es/workshops/recsystel2010/datatel.htm/` (2010) [Online; accedido 15 de Agosto de 2018].

[60] Stefaner, M., Dalla Vecchia, E., Condotta, M., Wolpers, M., Specht, M., Apelt, S., Duval, E.: Mace–enriching architectural learning objects for experience multiplication. In: European Conference on Technology Enhanced Learning, Springer (2007) 322–336

[61] Vuorikari, R., Massart, D.: Datatel challenge: European schoolnet's travel well dataset. In: 1st Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2010). (2010)

[62] Jack, K., Hammerton, J., Harvey, D., Hoyt, J.J., Reichelt, J., Henning, V.: Mendeleys reply to the datatel challenge. Procedia Computer Science **1** (2010) 1–3

[63] Niemann, K., Wolpers, M.: Usage context-boosted filtering for recommender systems in tel. In: European Conference on Technology Enhanced Learning, Springer (2013) 246–259

[64] Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Recommender systems handbook. Springer (2011) 257–297

[65] Buczak, A., Zimmerman, J., Kurapati, K.: Personalization: Improving ease-of-use, trust and accuracy of a tv show recommender. (2002)

[66] Bennett, J., Lanning, S., et al.: The netflix prize. In: Proceedings of KDD cup and workshop. Volume 2007., New York, NY, USA (2007) 35

[67] Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. information retrieval **4** (2001) 133–151

[68] Erdt, M., Fernandez, A., Rensing, C.: Evaluating recommender systems for technology enhanced learning: a quantitative survey. IEEE Transactions on Learning Technologies **8** (2015) 326–344

[69] Suskie, L.: Assessing student learning: A common sense guide. John Wiley & Sons (2018)

[70] Kuzilek, J., Hlosta, M., Zdrahal, Z.: Open university learning analytics dataset. Scientific data **4** (2017) 170171

[71] Lanzarini, L., Charnelli, M.E., Diaz, J.: Academic performance of university students and its relation with employment. In: Computing Conference CLEI, 2015 Latin American. (2015) 1–6

[72] Lanzarini, L., Charnelli, M.E., Baldino, G., Diaz, J.: Seleccion de atributos representativos del avance academico de los alumnos universitarios usando tecnicas de visualizacion: Un caso de estudio. Revista TE&ET (2015) 42–50

[73] Charnelli, M.E., Lanzarini, L., Díaz, J., Bariviera, A.: New item recommendation method based on latent topic extraction. In: Third Conference on Business Analytics in Finance and Industry (BAFI), ISCI (2018) 29